

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

Кафедра автоматизованих систем обробки інформації і управління

УДК: 004.89

«До захисту допущено»

В.о. завідувача кафедри

О.А.Павлов
(ініціали, прізвище)

(підпис)

“ ” 2019 р.

Дипломний проект
на здобуття ступеня бакалавра

з напрямку підготовки 6.050101 «Комп'ютерні науки»

на тему: «Комплекс задач кластеризації даних місцевих фінансів»

Виконала:

студентка 4 курсу, групи ІС-51

Коткова Ангеліна Андріївна

(прізвище, ім'я, по батькові)

(підпис)

Керівник

старший викладач Олійник Ю.О.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

**Консультант з
графічної
документації**

доц., к.т.н., доц. Тєлишева Т.О.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Рецензент

доц. каф. ТК, к.т.н., доц. Пасько В.П.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Засвідчую, що у цьому дипломному проекті
немає запозичень з праць інших авторів без
відповідних посилань.

Студентка Коткова А.А.

(підпис)

Київ – 2019 року

АНОТАЦІЯ

Пояснювальна записка дипломного проекту складається з шести розділів, 31 малюнків, 34 таблиць, 1 додатку, 31 джерела.

Дипломний проект присвячений розробці комплексу задач кластеризації даних місцевих фінансів. Для цього аналізуються такі дані: з чого складається, структура бюджетного середовища, бюджетна класифікація. На основі цих даних відбувається візуалізація, формується структура бюджету за доходами й витратами та динаміка бюджету. У результаті агрегації та обробки даних накоплюється інформація для наступного аналізу та візуалізації. У якості аналізу даних вирішується задача кластеризації територіальних бюджетів.

Метою роботи є покращення розподілу місцевих бюджетів на різні групи шляхом кластеризації даних місцевих фінансів.

У розділі інформаційного забезпечення було описано вхідні і вихідні дані, структуру бази даних.

Розділ математичного забезпечення присвячений вивченню змістовної та математичної постановки задачі, обґрунтуванню вибраних методів кластеризації та опису методів розв'язання.

У розділі з описом програмного забезпечення було визначено засоби розробки, вимоги до технічного забезпечення та представлено архітектуру програмного забезпечення.

У технологічному розділі наведено керівництво користувача, випробування програмного продукту.

КЛЮЧОВІ СЛОВА: МІСЦЕВІ БЮДЖЕТИ, ДОХОДИ, ВИДАТКИ, ДИНАМІКА БЮДЖЕТУ, КЛАСТЕРИЗАЦІЯ.

					КПІ.ІС-5113.1181-с.ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

ABSTRACT

The explanatory note of the diploma project consists of six sections, contains 31 drawings, 34 tables, 1 application and 31 sources.

The diploma project is devoted to the development of a set of tasks of clustering of data of local finances. For this purpose, the following data are analyzed: what is the composition of the budget environment, budget classification. On the basis of this data, visualization takes place, the structure of the budget for revenue and expenditure and the budget dynamics are formed. As a result of aggregation and data processing information is collected for further analysis and visualization. As the data analysis, the task of clustering territorial budgets is solved.

The aim of the work is to improve the distribution of local budgets to different groups by clustering local finance data.

In the information support section, the input and output data, the database structure were described.

The action of mathematical support is devoted to the definition of the content and mathematical formulation of the problem, the reasoning for the method of solution and the description of the solution methods.

The Software Description Section identified development tools, technical requirements and software architecture.

The technology section provides user guides, software product tests.

KEYWORDS: LOCAL BUDGETS, INCOME, EXPENDITURE, BUDGET DYNAMICS, CLUSTERIZATION.

ВСТУП

Для кожної держави бюджетна система є обов'язковою складовою. Державний бюджет виникає тоді, коли відбувається створення держави. Основою формування БСУ є Конституція України та Бюджетний кодекс України.

Розглянемо коротко історію формування відкритої фінансової системи. У результаті постійних політичних та економічних трансформацій виникло поняття «бюджетної політики», а згодом і «бюджету». Термін «бюджет» почали використовувати в 17 столітті. У ті часи відбувалося багато війн, а тому ускладнювався процес перерозподілу земель та контролю над надходженнями чи витратами коштів. Спершу розголошення даних по фінансуванню певних територій було заборонено. У кінці 19 – на початку 20 століття фінансові відносини виділили в окрему газуль наукової економіки. Чимало дослідників почали активно публікувати свої праці, а згодом у 2015 році в Україні почали запускати цілі портали, де публікуються дані по місцевих фінансах.

Саме тому задача дослідження місцевих фінансів є дуже актуальною. Є великі масиви даних про бюджетні показники. Їх можна переглянути на наступних порталах:

- Spending;
- Відкритий бюджет – бета;
- Офіційний портал Києва;
- Відкритий бюджет Херсонської області;
- Відкритий бюджет Кривого Рогу.

Необхідно розробити програмне забезпечення для інтелектуального аналізу цих даних. Однією з найважливіших задач є сегментація даних для знаходження схожих бюджетів за своїми характеристиками.

Наведемо основні визначення, що є важливими для теми кластеризації даних місцевих фінансів.

Доходи бюджету— це збільшення економічних вигод протягом звітного періоду у формі припливу або зростання активів, або зменшення зобов'язань, що спричиняють збільшення капіталу і не є внесками акціонерів [9].

Видатки бюджету – кошти, що спрямовуються на здійснення програм та заходів, передбачених відповідним бюджетом, за винятком коштів на погашення основної суми боргу та повернення надміру сплачених до бюджету сум [3].

Дипломний проект присвячений розробці комплексу задач кластеризації даних місцевих фінансів.

Цільовою аудиторією даної розробки є спеціалісти фінансових органів місцевих бюджетів.

Практичне значення одержаних результатів. Розробка задачі кластеризації даних місцевих бюджетів для покращення розподілу місцевих бюджетів на різні групи. Для бюджетів зі «схожими» характеристиками можна підбирати однакові чи подібні програми розвитку, що значно полегшить роботу спеціалістам, що працюють в органах місцевих бюджетів.

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

1 ЗАГАЛЬНІ ПОЛОЖЕННЯ

1.1 Опис предметного середовища

1.1.1 Опис предметної області

Для кожної держави бюджетна система є обов'язковою складовою. Розглянемо бюджетну систему України (БСУ). Основою формування БСУ є Конституція України та Бюджетний кодекс України.

Україна, згідно з Конституцією, - унітарна держава. Це означає, що країна має дві ланки: державний бюджет та багаточисельні місцеві бюджети. Систему адміністративно-територіального устрою України складають: Автономна Республіка Крим, області, райони, міста, райони в містах, селища і села [11], тобто 25 адміністративно-територіальних одиниць, що включають Автономну Республіку Крим та 24 області.

1. Згідно зі ст.2 Бюджетного кодексу України бюджетна система України – це сукупність державного бюджету та місцевих бюджетів, побудована з урахуванням економічних відносин, державного та адміністративно-територіальних устроїв і врегульована нормами права [2].

БСУ згідно зі статтю 5 Бюджетного кодексу України складається з державного бюджету та місцевих бюджетів. До місцевих бюджетів належать:

- бюджет Автономної Республіки Крим;
- обласні;
- районні бюджети;
- бюджети місцевого самоврядування.

До бюджетів місцевого самоврядування, у свою чергу, відносяться:

- бюджети територіальних громад сіл та їх об'єднань,
- бюджети селищ,
- бюджети міст (у тому числі районів у містах),

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

– бюджети об'єднаних територіальних громад, що створюються згідно із законом та перспективним планом формування територій громад.

Структуру БСУ схематично наведено на рисунку 1.1.

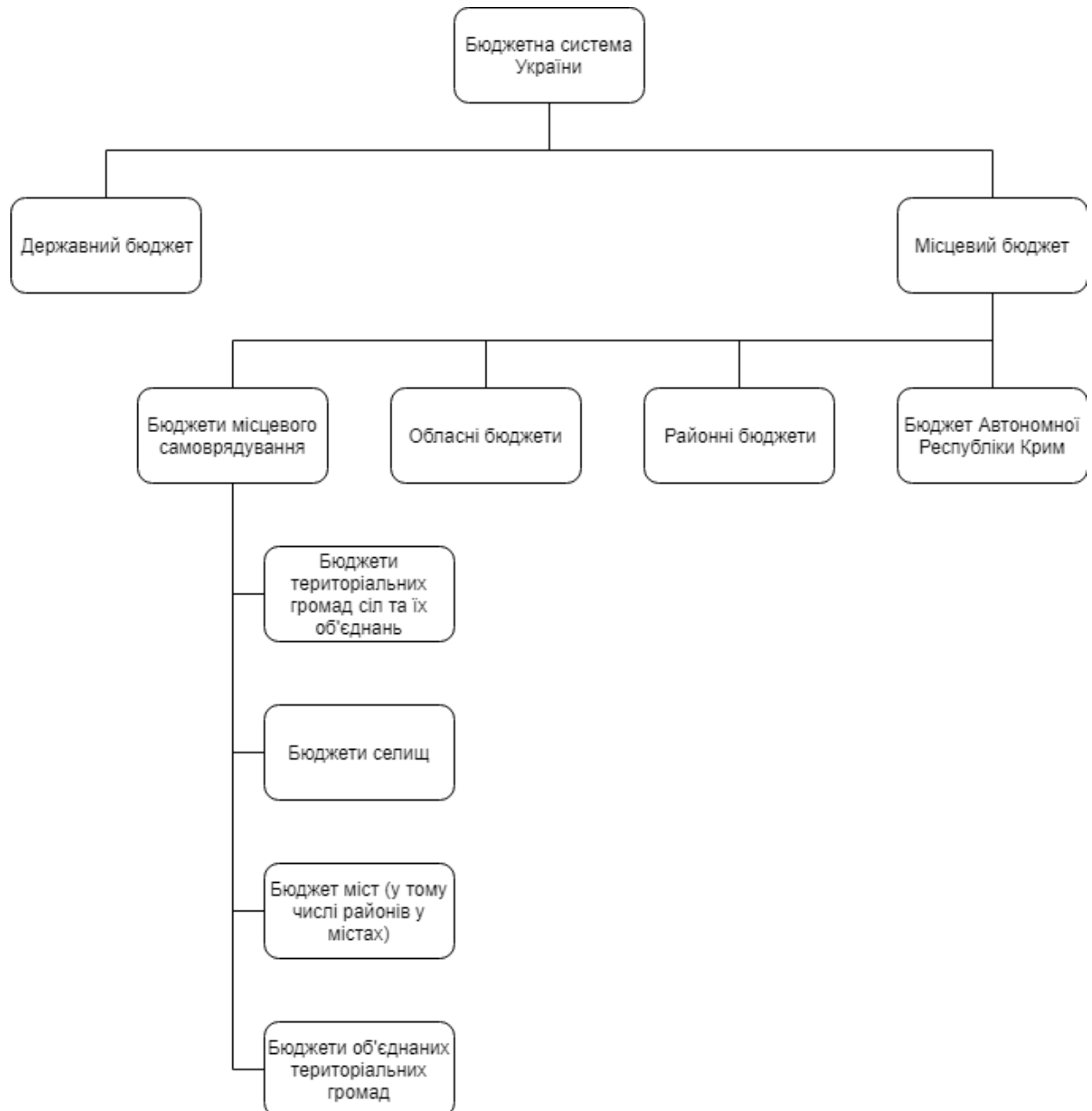


Рисунок 1.1 – Структура БСУ

Згідно зі статтею 7 Бюджетного кодексу України БСУ ґрунтується на наступних принципах: єдність, збалансованість, самостійність, повнота, обґрунтованість, ефективність та результативність, субсидіарність, цільове

використання бюджетних коштів, справедливість та неупередженість, публічність та прозорість.

1.1.2 Опис процесу діяльності

Запропонований програмний продукт присвячений роботі з даними місцевих бюджетів. Опис процесу діяльності представлено у вигляді трьох незалежних процесів.

Перший процес – завантаження даних по обраному місцевому бюджету (рисунок 1.2). Клієнт здійснює вибір місцевого бюджету, портал надає дані по місцевому бюджету, програмне забезпечення реагує на запит клієнта та здійснює завантаження даних по місцевому бюджету. У результаті клієнт отримує повідомлення про завантаження даних по обраному місцевому бюджету.

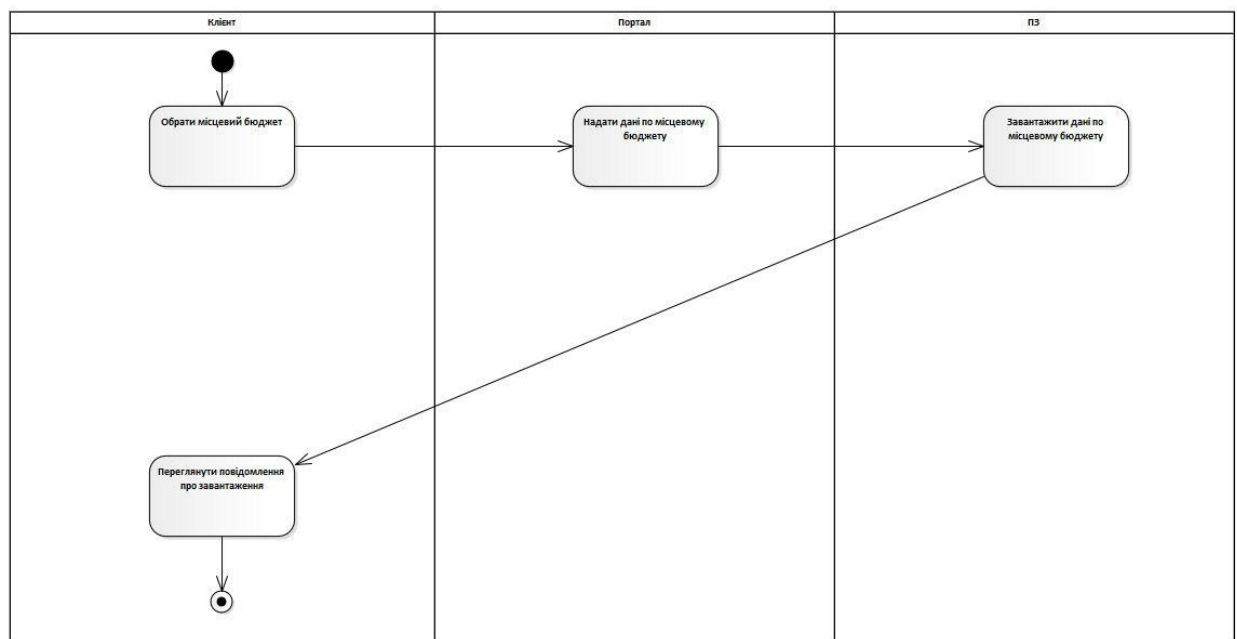


Рисунок 1.2 – Activity діаграма для завантаження даних

Другий процес – візуалізація даних залежно від обраного параметру візуалізації (рисунок 1.3). Клієнт обирає, за яким параметром візуалізувати дані, а саме: формування за доходами, формування за витратами чи формування динаміки бюджету в цілому. Під динамікою бюджету розуміємо

графічне представлення доходів та витрат місцевого бюджету в розрізі місяців року. Портал, у свою чергу, надає візуалізовані дані залежно від обраного параметру, після чого клієнт може переглянути візуалізацію.

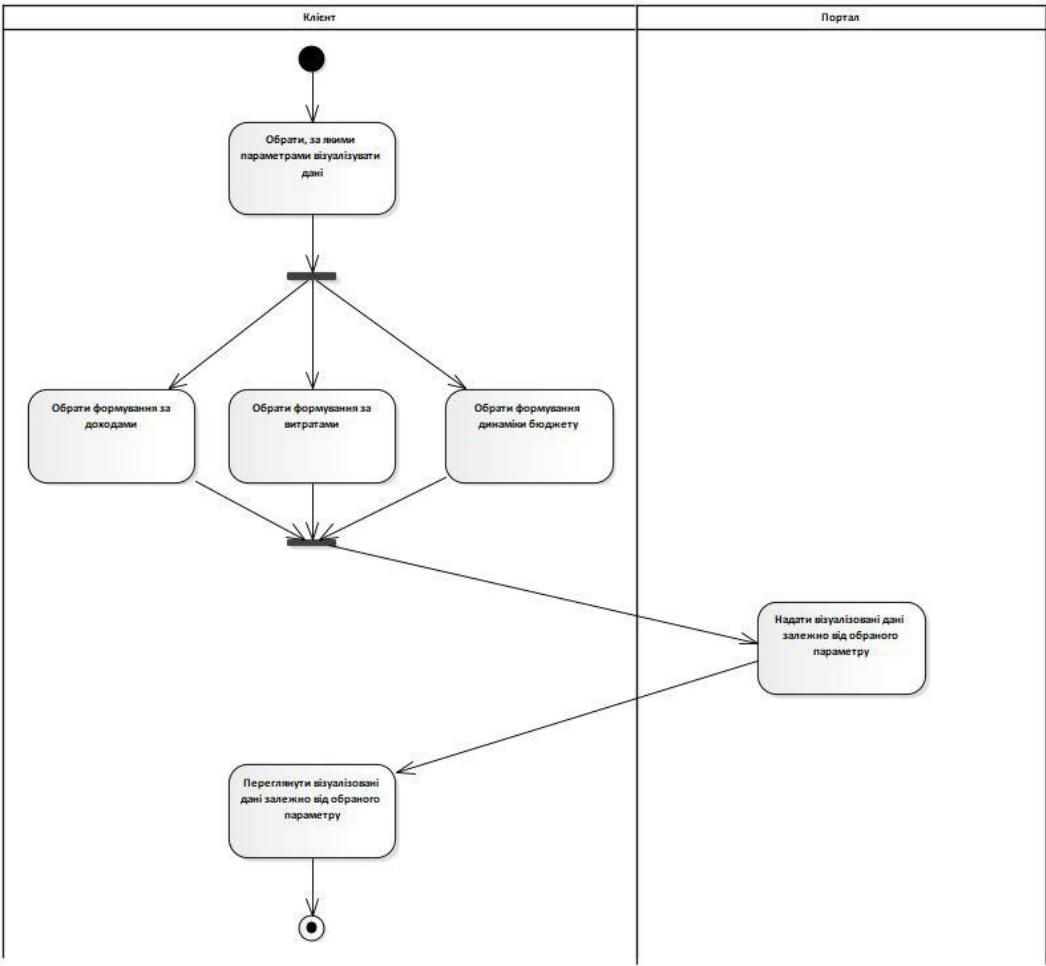


Рисунок 1.3 – Activity діаграма для візуалізації даних

Третій процес – кластеризація місцевих бюджетів (рисунок 1.4). Під кластеризацією бюджету розуміємо розподіл місцевих бюджетів на різні групи на основі даних про структуру доходів та витрат бюджету.

Клієнт обирає параметр кластеризації місцевого бюджету, після цього програмне забезпечення проводить процедуру кластеризації. У результаті користувач може переглянути результати кластеризації.

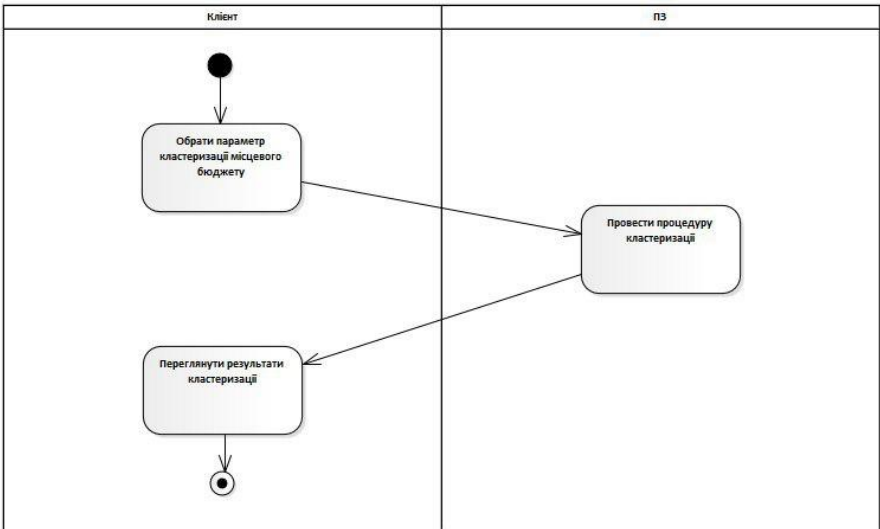


Рисунок 1.4 – Activity діаграма для кластеризації

1.1.3 Опис функціональної моделі

Специфікацію функціональної поведінки системи представимо у вигляді діаграми варіантів використання. На рисунку 1.5 наведено всі функції системи та описано акторів, які будуть їх використовувати.

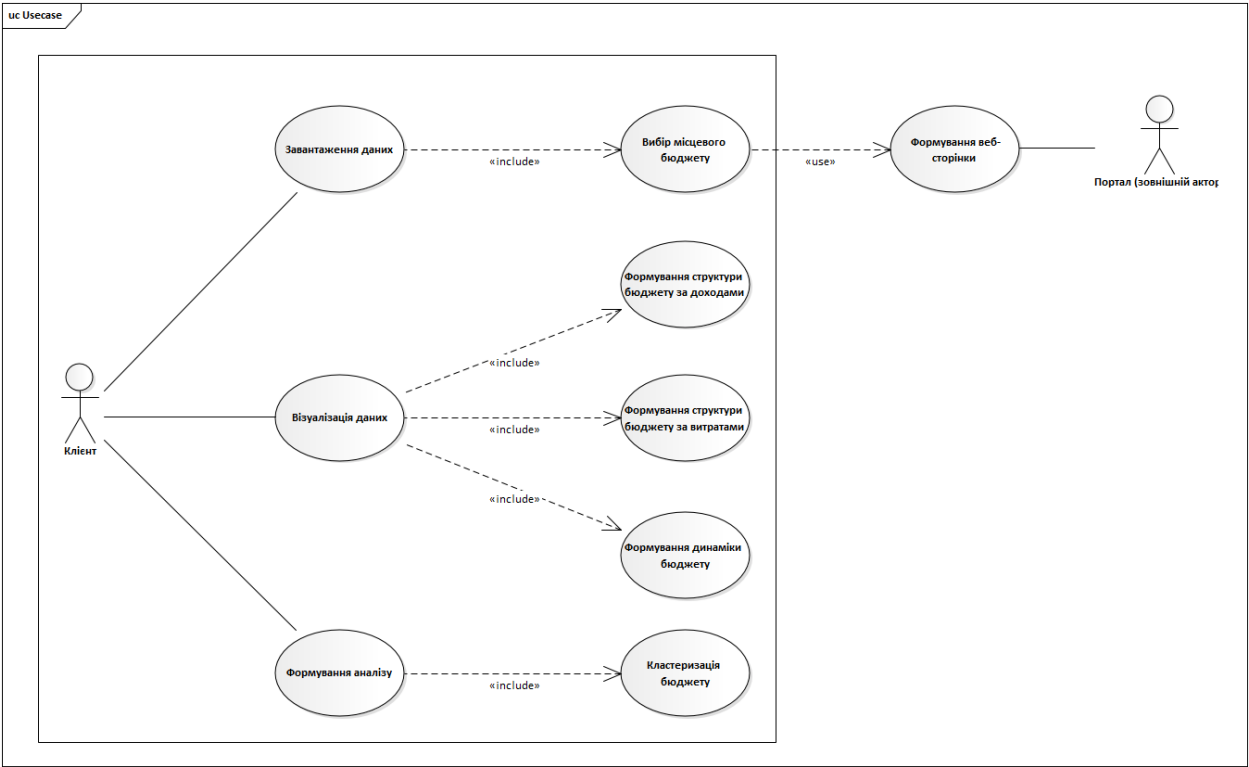


Рисунок 1.5 – Схема структурна варіантів використання

Безпосередньо із системою будуть взаємодіяти два актори:

- клієнт – це особа, яка може працювати із основними функціями системи;
- портал (зовнішній клієнт) – це наша система, яка формує веб-сторінки.

Відповідно до визначених варіантів використання виявлено функціональні вимоги та встановлено їх пріоритетність. Результат наведено в таблиці 1.1.

Таблиця 1.1 – Функціональні вимоги

Варіант використання	Функціональна вимога	Пріоритет
Вибір місцевого бюджету	1. Система надає можливість вибору конкретного місцевого бюджету	Високий
Завантаження даних	1.1. Система надає можливість завантаження даних з порталу openbudget.gov.ua	Високий
Візуалізація даних	2. Система надає можливість візуалізувати дані за вибраним місцевим бюджетом	Високий
Формування структури бюджету за доходами	2.1. Система надає можливість сформувати структуру бюджету за обраним місцевим бюджетом	Середній
Формування структури бюджету за витратами	2.2. Система надає можливість сформувати структуру бюджету за витратами за обраним місцевим бюджетом	Середній
Формування динаміки бюджету	2.3. Система надає можливість сформувати динаміку бюджету за обраним місцевим бюджетом	Середній
Формування аналізу	3. Система надає можливість сформувати аналіз за обраним місцевим бюджетом	Високий
Кластеризація бюджету	3.1. Система надає можливість кластеризувати бюджети	Високий

1.2 Огляд наявних аналогів

Перед початком роботи над дипломним проектом було здійснено пошук застосунків зі схожою функціональністю.

Нині існують програмні продукти, які здійснюють огляд місцевих фінансів. У таблиці 1.2 наведено назви наявних аналогів та режими доступів до них.

Таблиця 1.2 – Наявні аналоги

Назва	Режим доступу
Spending	https://spending.gov.ua/
Відкритий бюджет – бета	http://openbudget.in.ua/
Офіційний портал Києва	https://kyivcity.gov.ua/finansy_ta_biudzhethet/vidkrytyi_biudzhethet_Shortcut_227363.html
Відкритий бюджет Херсонської області	https://openbudget.ks.ua/
Відкритий бюджет Кривого Рогу	https://openbudget.krmisto.gov.ua/

Розглянемо детальніше кожен із наведених аналогів.

– Spending [22]

На рисунку 1.6 наведено логотип порталу «Spending».



Рисунок 1.6 – Логотип порталу «Spending»

Spending – це веб-портал, основним призначенням якого є облік використання публічних коштів. Цей інформаційний портал підтримує ідею «Прозорого бюджету». Метою проекту є створення відкритого ресурсу, що надає інформацію про державні фінанси. Портал функціонує з 15 вересня 2015 року. Spending є одним із основних модулів проекту Міністерства фінансів України «Є-data» [23].

Основною інформацією, що оприлюднюється на порталі є:

1) транзакції казначейства;

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

2) звітність та договори розпорядників, державних цільових фондів, державних та комунальних підприємств.

– Відкритий бюджет – бета [5]

На рисунку 1.7 наведено логотип порталу «Відкритий бюджет – бета».



Рисунок 1.7 – Логотип порталу «Відкритий бюджет – бета»

Open Budget – це веб-портал, що призначений для відслідковування:

а) бюджету, а саме: доходів, видатків, кредитування, фінансування та державних фондів;

б) боргу, а саме: стану боргу та стану гарантованого боргу;

в) місцевих бюджетів, а саме: доходів та видатків.

«Відкритий бюджет – бета», або «Бюджет для громадськості» вважається складовою другого модулю проекту Міністерства фінансів України «Є-data». Він є інструментом управління публічними коштами.

– Офіційний портал Києва [12]

На рисунку 1.8 наведено логотип порталу «Офіційного порталу Києва».



Рисунок 1.8 – Логотип порталу «Офіційного порталу Києва»

Офіційний портал киева – це веб-застосунок, що призначений для публікування інформації про події, що відбуваються в Києві, а саме: про міську владу та сервіси, громадськості, новини та про відкритий бюджет міста Києва.

У вкладці Фініси та бюджети – Бюджет наведено графічне представлення видатків та доходів з 2014 до 2019 року помісячно. Тут можна

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

відслідкувати фактичні та заплановані доходи та витрати бюджету, а також переглянути відсотки виконання бюджетів окремо по доходах та видаткам.

– Відкритий бюджет Херсонської області [7]

На рисунку 1.9 наведено логотип порталу «Відкритого бюджету Херсонської області».



Рисунок 1.9 – Логотип порталу «Відкритого бюджету Херсонської області»

Сайт «Відкритого бюджету Херсонської області» призначений для того, щоб кожен користувач міг переглядати дані про бюджет у простій та доступній формі. Так, людина, яка навіть не є спеціалістом у економіці, може легко переглядати суми доходів та видатків з їх відповідним графічним представленням, що допомагає сформувати думку щодо формування бюджету та витрачання бюджетних коштів у Херсонській області.

– Відкритий бюджет Кривого Рогу [6]

На рисунку 1.10 наведено логотип порталу «Відкритого бюджету Кривого Рогу».



Рисунок 1.10 – Логотип порталу «Відкритого бюджету Кривого Рогу»

Сайт «Відкритий бюджет Кривого Рогу» – це веб-портал, що призначений для відображення та аналізу бюджету Кривого Рогу. Тут користувач може переглянути інформацію про бюджет міста за 2016, 2017 та 2018 роки, ознайомитися із структурою бюджету міста, переглянути детальну інформацію про міський бюджет, а саме: податкові надходження, офіційні

трансфери, неоподатковані надходження, цільові фонди, місцеві запозичення до бюджету розвитку, доходи від операцій з капіталом. На сайті також міститься загальна інформація про районні бюджети Кривого Рогу, а саме: надходження та витрати.

Проаналізуємо функції, які виконують зазначені вище програмні продукти. Дані по функціям наведено в таблиці 1.3.

Таблиця 1.3 – Огляд функцій наявних аналогів

Аналоги Функції	Spending	Відкритий бюджет - бета	Офіційний портал Києва	Відкритий бюджет Херсонськ ої області	Відкритий бюджет Кривого Рогу
Візуалізація доходів	—	+	+	+	+
Візуалізація витрат	—	+	+	+	+
Формування графіків	—	+	+	+	—
Деталізація до трансакцій	+	—	—	—	—
Візуалізація на мапі	+	+	—	—	+
Кластеризація бюджету	—	—	—	—	—

Із таблиці 1.3 видно, що ті аналоги, які представлені нині на ринку, не охоплюють увесь спектр функцій, які запропоновані в даному дипломному проекті. До того ж для аналізу даних місцевих бюджетів ці застосунки не використовують кластеризацію, що є ключовим для даної роботи.

1.3 Постановка задачі

1.3.1 Призначення розробки

Запропонований програмний продукт призначений для аналізу місцевих бюджетів. Для аналізу є дані місцевих бюджетів: з чого складається, структура бюджетного середовища, бюджетна класифікація. На основі цих даних відбувається візуалізація, формується структура бюджету за

доходами та витратами, а також динаміка бюджету. У результаті відбувається формування аналізу, а саме – кластеризація бюджету.

1.3.2 Цілі та задачі розробки

У Data Mining часто легше виділити групи схожих об'єктів, у нашому випадку – це схожі місцеві бюджети, вивчити їх особливості та навести аналіз для кожної групи окремо, ніж створювати одну загальну модель для всіх даних.

Основна мета роботи – покращення розподілу місцевих бюджетів на різні групи шляхом кластеризації даних місцевих фінансів.

Для реалізації поставленої мети необхідно розв'язати наступні задачі.

- 1) Завантаження даних із порталу openbudget.gov.ua.
- 2) Завантаження даних до бази даних.
- 3) Створення засобів візуалізації даних.
- 4) Створення засобів аналізу даних (кластеризація даних місцевих бюджетів).

Висновок до розділу

У розділі загальних положень було розглянуто предметне середовище, наведено структуру бюджетної системи України та її основні принципи. Дипломний проект присвячений роботі з даними місцевих бюджетів.

При описі предметного середовища було визначено та описано процеси діяльності, що представлені у вигляді трьох незалежних процесів, для кожного з яких наведено структурні схеми. Також було створено функціональну модель системи, а саме: побудовано USE-case діаграму, описано акторів системи та визначено функціональні вимоги з пріоритетами виконання відповідно до варіантів використання.

Наступним етапом роботи над дипломним проектом став пошук аналогів запропонованого застосунку з подібним функціоналом. Було порівняно функції, які виконують знайдені програмні продукти. На даний момент було виявлено декілька систем зі схожим функціоналом, але вони не охоплюють увесь спектр функцій, які запропоновані в даному дипломному проекті. До того ж для аналізу даних місцевих бюджетів ці застосунки не використовують кластеризацію, що є ключовим для даної роботи.

Також у одному з підрозділів було сформовано постановку задачі, визначено призначення, мету та задачі розробки.

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Вхідні дані

Вхідними документами комплексу задач є:

- рішення сесії про затвердження місцевого бюджету;
- розпис місцевого бюджету.

У таблиці 2.1 наведено детальний опис вхідних документів для розпису місцевого бюджету.

Таблиця 2.1 – Опис вхідних документів

№	Найменування	Реквізити
1.	Розпис доходів місцевих бюджетів	Тип фонду, код, найменування коду, розпис на рік, кошторис на рік, виконано всього, відсоток виконання
2.	Розпис витрат місцевих бюджетів	Тип фонду, КПК, найменування КПК, КФК, найменування КФК, КЕКВ, найменування КЕКВ, розпис на рік, кошторис на рік, виконано всього, відсоток виконання

Розшифруємо абривіатури, що було використано для опису вхідних документів:

- КПК – код програмної класифікації,
- КФК – код функціональної класифікації,
- КЕКВ – «Коди економічної класифікації витрат».

Документи з детальним описом місцевих бюджетів мають формат .json. Для кожної області було завантажено дані для населених пунктів окремо для доходів та витрат.

Фрагмент документу з даними по доходам наведено на рисунку 2.1. Кожне поле містить наступні атрибути:

- fundType – це тип фонду;
- incomeCode – це код доходу;

- incomeCodeName – це найменування доходу;
- yearBudgetPlan – це розпис на рік (у грн);
- yearBudgetEstimate – це кошторис на рік (у грн);
- totalDone – це скільки виконано всього (у грн);
- percentDone – це відсоток виконання (показує відношення всього виконаного до розпису на рік).

```
{
  "fundType": "COMMON",
  "incomeCode": "10000000",
  "incomeCodeName": "Податкові надходження",
  "yearBudgetPlan": 678550,
  "yearBudgetEstimate": null,
  "totalDone": 40008.19,
  "percentDone": 5.89
},
{
  "fundType": "COMMON",
  "incomeCode": "13000000",
  "incomeCodeName": "Рентна плата та плата за використання інших природних ресурсів",
  "yearBudgetPlan": 7200,
  "yearBudgetEstimate": null,
  "totalDone": 0,
  "percentDone": 0
},
{
  "fundType": "COMMON",
  "incomeCode": "13010000",
  "incomeCodeName": "Рентна плата за спеціальне використання лісових ресурсів",
  "yearBudgetPlan": 7200,
  "yearBudgetEstimate": null,
  "totalDone": 0,
  "percentDone": 0
},
}
```

Рисунок 2.1 – Приклад вхідного файлу по доходам

Фрагмент документу з даними по витратам наведено на рисунку 2.2.

Кожне поле містить наступні атрибути:

- fundType – це тип фонду;
- programCode – це код КПК;
- programCodeName – це найменування КПК;
- functionCode – це код КФК;
- functionCodeName – це найменування КФК;
- economicCode – це код КЕКВ;
- economicCodeName – це найменування КЕКВ;
- yearBudgetPlan – це розпис на рік (у грн);

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

- yearBudgetEstimate – це кошторис на рік (у грн);
- totalDone – це скільки виконано всього (у грн);
- percentDone – це відсоток виконання (показує відношення всього виконаного до розпису на рік).

```
{
  "fundType": "COMMON",
  "programCode": "0150",
  "programCodeName": null,
  "functionCode": null,
  "functionCodeName": null,
  "economicCode": "2111",
  "economicCodeName": "Заробітна плата",
  "yearBudgetPlan": 420366,
  "yearBudgetEstimate": 420366,
  "totalDone": 28515.6,
  "percentDone": 6.78
},
{
  "fundType": "COMMON",
  "programCode": "0150",
  "programCodeName": null,
  "functionCode": null,
  "functionCodeName": null,
  "economicCode": "2120",
  "economicCodeName": "Нарахування на оплату праці",
  "yearBudgetPlan": 98053,
  "yearBudgetEstimate": 98053,
  "totalDone": 6682.97,
  "percentDone": 6.81
},
}
```

Рисунок 2.2 – Приклад вхідного файлу по витратам

Для проведення кластеризації було створено підготовчий документ у форматі .xsl. Розглянемо цей документ на прикладі доходів. Фрагмент документу наведено на рисунку 2.3.

Sum of розпис на рік	Column Labels							
Row Labels	10000000	13000000	13010000	13010200	14000000	14040000	18000000	18010000
18307519000	1783010	6000	6000	6000	1632	1632	1775378	1195480
18308523000	2126000	10000	10000	10000	60000	60000	2056000	1094000
Grand Total	3909010	16000	16000	16000	61632	61632	3831378	2289480

Рисунок 2.3 – Приклад вхідного підготовчого документу для проведення кластеризації для доходів

У цьому документі міститься вся інформація по кожному населеному пункту кожної області у зведеному вигляді.

Горизонтальні поля – це коди доходів, вертикальні поля – це коди населених пунктів. Основна інформація – це суми розпису на рік, кошторису на рік та суми виконання залежно від обраного фільтру.

2.2 Вихідні дані

Вихідними даними для даного програмного продукту є карти Кохонена, які коистувач може переглядати. Завдяки картам можна знайти приховані закономірності в даних, ґрунтуючись на зниженні розмірності вихідного простору в простір меншої розмірності.

У нашому випадку аналіз бюджету, а саме – кластеризацію, візуалізовано саме таким способом.

2.3 Опис структури бази даних

Для даного дипломного проекту було прийнято рішення використовувати нереляційну базу даних (БД) MongoDB. Схему БД для збереження вхідних файлів наведено на рисунку 2.4.

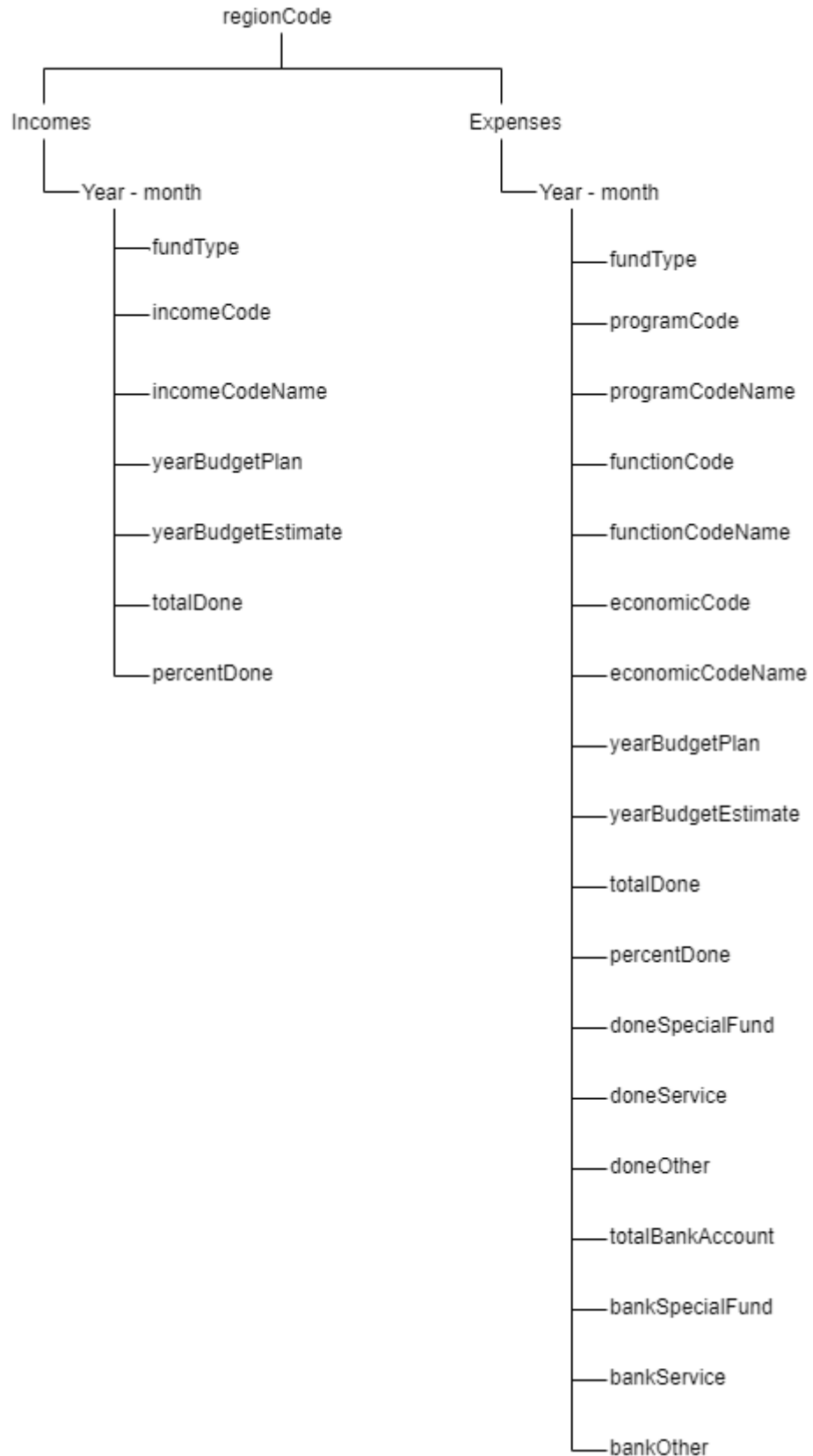


Рисунок 2.4 – Схема структур даних

Змн.	Арк.	№ докум.	Підпис	Дата

Щоб зробити правильну структуру, що підходить до даного проекту було використано скрипт, написаний мовою JavaScript, завдяки якому було розбито всі файли з розширенням .json та внесено їх до БД у правильному порядку. Скрипт працює за допомогою технології Node.js. Ця технологія використовується для написання серверної частини веб-застосунків.

На рисунку 2.5 наведено структуру БД.

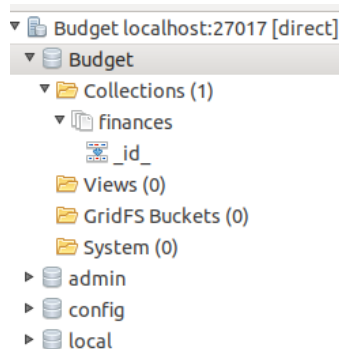


Рисунок 2.5 – Структура БД

На рисунку 2.6 наведено вміст колекції finances.

_id	regionCode	incomes	expenses	_v
5cdedb150f13dc085dd94673	09304501000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd9468b	09304502000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd946a3	09304503000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd946bb	09304504000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd946d3	09304301000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd946eb	09304401000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd94703	03308502000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd9471b	03308505000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd94733	03308503000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd9474b	03308507000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd94763	03308508000	[12 elements]	[11 elements]	0
5cdedb150f13dc085dd9477b	03308200000	[12 elements]	[11 elements]	0

Рисунок 2.6 – Вміст колекції finances

Детальний опис атрибутів колекції finances наведено в таблиці 2.2.

Таблиця 2.2 – Опис колекції finances

Атрибут	Пояснення	Тип
_id	Унікальний код кожного масиву об'єктів	ObjectId
regionCode	Код кожної області	String
incomes	Масив з даними по доходам	Array
experiences	Масив з даними по видаткам	Array

На рисунку 2.7 наведено вміст масиву об'єктів incomes.

Document id	_id	year	month	budgetData
Scdedb150f13dc085dd9468b	Scdedb150f13	2018	1	[60 elements]
Scdedb150f13dc085dd946a3	Scdedb150f13	2018	1	[50 elements]
Scdedb150f13dc085dd946bb	Scdedb150f13	2018	1	[47 elements]
Scdedb150f13dc085dd946d3	Scdedb150f13	2018	1	[99 elements]
Scdedb150f13dc085dd946eb	Scdedb150f13	2018	1	[45 elements]
Scdedb150f13dc085dd94703	Scdedb150f13	2018	1	[38 elements]
Scdedb150f13dc085dd9471b	Scdedb150f13	2018	1	[74 elements]
Scdedb150f13dc085dd94733	Scdedb150f13	2018	1	[58 elements]
Scdedb150f13dc085dd9474b	Scdedb150f13	2018	1	[57 elements]
Scdedb150f13dc085dd94763	Scdedb150f13	2018	1	[60 elements]
Scdedb150f13dc085dd9477b	Scdedb150f13	2018	1	[78 elements]
Scdedb150f13dc085dd94793	Scdedb150f13	2018	1	[74 elements]

Рисунок 2.7 – Вміст масиву incomes

Детальний опис атрибутів масиву incomes наведено в таблиці 2.3.

Таблиця 2.3 – Опис масиву incomes

Атрибут	Пояснення	Тип
_id	Унікальний код кожного об'єкту	ObjectId
year	Рік	String
month	Місяць	String
budgetData	Масив з даними по документам з доходами	Array

На рисунку 2.8 наведено вміст об'єкту budgetData масиву incomes.

Document id	fundType	incomeCode	incomeCodeName	yearBudgetPlan	yearBudgetEstimate	totalDone	percentDone
Scdedb150f13dc085dd94673	COMMON	14000000	Внутрішні податки на товари	7000	null	7652	109.31
Scdedb150f13dc085dd9468b	COMMON	13000000	Рентна плата та плата за вико	780	null	1429.52	183.27
Scdedb150f13dc085dd946a3	COMMON	13000000	Рентна плата та плата за вико	0	null	-350	null
Scdedb150f13dc085dd946bb	COMMON	14000000	Внутрішні податки на товари	2000	null	4633.5	231.67
Scdedb150f13dc085dd946d3	COMMON	11000000	Податки на доходи, податки	0	null	3656	null
Scdedb150f13dc085dd946eb	COMMON	14000000	Внутрішні податки на товари	30000	null	33701.44	112.33
Scdedb150f13dc085dd94703	COMMON	14000000	Внутрішні податки на товари	11755	null	14182	120.64
Scdedb150f13dc085dd9471b	COMMON	11000000	Податки на доходи, податки	0	null	340	null
Scdedb150f13dc085dd94733	COMMON	14000000	Внутрішні податки на товари	7320	null	8662	118.33
Scdedb150f13dc085dd9474b	COMMON	11000000	Податки на доходи, податки	0	null	340	null
Scdedb150f13dc085dd94763	COMMON	14000000	Внутрішні податки на товари	23050	null	17482.4	75.84
Scdedb150f13dc085dd9477b	COMMON	11000000	Податки на доходи, податки	117402168	null	100178724.11	85.32

Рисунок 2.8 – Вміст об'єкту budgetData масиву incomes

Детальний опис атрибутів об'єкту budgetData масиву incomes наведено в таблиці 2.4.

Таблиця 2.4 – Опис атрибутів об'єкту budgetData масиву incomes

Атрибут	Пояснення	Тип
fundType	Тип фонду	String
incomeCode	Код доходу	String

Продовження таблиці 2.4

incomeCodeName	Найменування доходу	String
yearBudgetPlan	Розпис на рік	Int32
yearBudgetEstimate	Кошторис на рік	Int32
totalDone	Виконано всього	Double
percentDone	Відсоток виконання	Double

На рисунку 2.9 наведено вміст масиву об'єктів expenses.

{Document id}	_id	year	month	budgetData
Scdedb150f13dc085dd94673	Scdedb150f13dc085dd94688	2018	12	[75 elements]
Scdedb150f13dc085dd9468b	Scdedb150f13dc085dd946a0	2018	12	[55 elements]
Scdedb150f13dc085dd946a3	Scdedb150f13dc085dd946b8	2018	12	[60 elements]
Scdedb150f13dc085dd946bb	Scdedb150f13dc085dd946d0	2018	12	[44 elements]
Scdedb150f13dc085dd946d3	Scdedb150f13dc085dd946e8	2018	12	[163 elements]
Scdedb150f13dc085dd946eb	Scdedb150f13dc085dd94700	2018	12	[50 elements]
Scdedb150f13dc085dd94703	Scdedb150f13dc085dd94718	2018	12	[45 elements]
Scdedb150f13dc085dd9471b	Scdedb150f13dc085dd94730	2018	12	[78 elements]
Scdedb150f13dc085dd94733	Scdedb150f13dc085dd94748	2018	12	[84 elements]
Scdedb150f13dc085dd9474b	Scdedb150f13dc085dd94760	2018	12	[91 elements]
Scdedb150f13dc085dd94763	Scdedb150f13dc085dd94778	2018	12	[93 elements]
Scdedb150f13dc085dd9477b	Scdedb150f13dc085dd94790	2018	12	[597 elements]

Рисунок 2.9 – Вміст масиву об'єктів expenses

Детальний опис масиву expenses наведено в таблиці 2.5.

Таблиця 2.5 – Опис масиву expenses

Атрибут	Пояснення	Тип
_id	Унікальний код кожного об'єкту	ObjectId
year	Рік	String
month	Місяць	String
budgetData	Масив з даними по документам з доходами	Array

На рисунку 2.10 наведено вміст об'єкту budgetData масиву expenses.

{Document id}	fundType	programC	programCodeN	functionCc	functionName	economicCode	economicCodeN	yearBudgetPlz	yearBudgetEs	totalDone	percentDone
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	847600	847600	805541.32	95.03
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	734500	734500	603070.91	82.1
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	565297.25	565297.25	535389.11	94.7
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	420000	420000	301512.55	71.78
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	2762700	2762700	1819207.81	65.84
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	936594	936594	812986.86	86.8
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	690226	690226	557046.22	80.7
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	648975	648975	561234.42	86.48
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	457288	457288	385769.92	84.36
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	850435	850435	606109.54	71.27
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	994325	994325	815230.31	81.98
Scdedb150f1	COMMON	0150	Організаційн	0111	Вищі органи дер	0000	null	3968058	3968058	2828761.2	71.28

Рисунок 2.10 – Вміст об'єкту budgetData масиву expenses

Детальний опис атрибутів об'єкту budgetData масиву expences наведено в таблиці 2.6.

Таблиця 2.6 – Опис атрибутів об'єкту budgetData масиву incomes

Атрибут	Пояснення	Тип
fundType	Тип фонду	String
programCode	Код КПК	String
programCodeName	Найменування КПК	String
functionCode	Код КФК	String
functionCodeName	Найменування КФК	String
economicCode	Код КЕКВ	String
economicCodeName	Найменування КЕКВ	String
yearBudgetPlan	Розпис на рік	Int32
yearBudgetEstimate	Кошторис на рік	Int32
totalDone	Виконано всього	Double
percentDone	Відсоток виконання	Double

Висновок до розділу

У розділі з описом інформаційного забезпечення було описано вхідні документи комплексу задач: рішення сесії про затвердження місцевого бюджету та розпис місцевого бюджету. Розпис місцевого бюджету включає в себе документи з даними по доходам та витратам місцевих бюджетів.

Документи з детальним описом місцевих бюджетів мають формат .json. Для кожної області було завантажено дані для населених пунктів окремо для доходів та витрат. Ще одним вхідним документом є підготовчий документ у форматі .xsl. Він створений для зручності проведення кластеризації, так як у цьому документі міститься зведена інформація по кожному населеному пункту та по кожному коду.

Для даного дипломного проекту було прийнято рішення використовувати нереляційну БД MongoDB. Для того, щоб зробити правильну структуру, що підходить до даного проекту було використано скрипт, написаний мовою JavaScript, завдяки якому було розбито всі файли з розширенням .json та внесено їх до БД у правильному порядку.

Також у розділі було наведено схему та структуру БД, детально розписано вміст колекції, а також масивів з об'єктами, що входять до неї.

3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Змістовна постановка задачі

Є країна, що складається з 25 адміністративно-територіальних одиниць. Для кожної адміністративно-територіальної одиниці зібрана інформація по доходам та витратам по всім населеним пунктам.

Для ефективного аналізу бюджету необхідно виконати кластеризацію за наступними показниками:

- розпис на рік;
- кошторис на рік;
- виконано всього.

Задачею кластеризації є розподілення об'єктів на «схожі» групи. Такі групи називають кластерами. Кластер – це група елементів, що є подібними між собою за певним критерієм [1].

У результаті такої постановки задачі можемо сформулювати мету.

Мета: необхідно віднести об'єкти до одного чи декількох кластерів. Усі об'єкти формуються одночасно, після чого розподіляються по кластерам.

Рішенням поставленої задачі є розподілення всіх об'єктів до відповідних кластерів.

3.2 Математична постановка задачі

Дано:

- файл по доходах;
- файл по видатках;
- екземпляри даних визначені числовими значеннями;
- клас є невідомим для конкретного екземпляру.

Знайти:

- міру подібності даних для оптимальної роботи методів;

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

– згідно з визначеною мірою подібності знайти найкращий спосіб кластеризації.

Запропоновану задачу можна записати формально наступним чином.

Дано: множина об'єктів I . Кожна множина об'єктів I представлена набором атрибутів.

Знайти: множину кластерів K . Для множини кластерів K справедливе наступне відношення: $F: I \rightarrow K$, де F – це відображення множини об'єктів I на множину кластерів K .

Множина $I = \{i_1, i_2, \dots, i_j, i_n\}$, де i_j – це об'єкт дослідження.

3.3 Обґрунтування методу розв'язання

Для реалізації поставленої задачі кластеризації даних місцевих фінансів можна виділити дві основні класифікації алгоритмів кластеризації (рисунок 3.1).

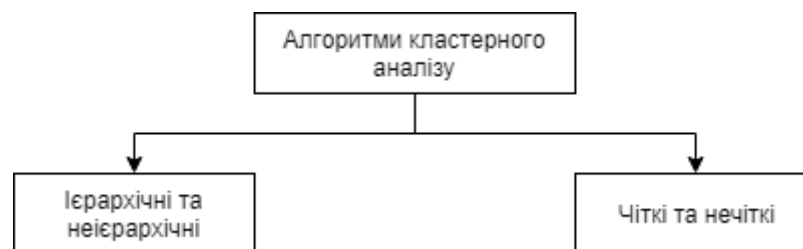


Рисунок 3.1 – Класифікація алгоритмів кластерного аналізу

Розглянемо детальніше кожен із наведених алгоритмів кластерного аналізу.

– Ієрархічні та неієрархічні:

1) ієрархічні алгоритми, або алгоритми таксомонії, являють собою систему вкладених розбиттів. На виході роботи такого алгоритму маємо дерево кластерів, у якого корінь – це вся вибірка, листя – найбільш дрібні кластери;

2) неієрархічні алгоритми являють собою одне розбиття вибірки на кластери.

– Чіткі та нечіткі:

1) чіткі, або непересічні, алгоритми працюють так, що кожному об'єкту ставиться у відповідність номер кластера. Тобто кожен об'єкт належить тільки одному кластеру;

2) нечіткі, або пересічні, алгоритми працюють так, що кожному об'єкту ставиться у відповідність набір речових значень. Це показує ступінь відношення об'єкта до кластерів. У цьому випадку об'єкт відноситься до кожного кластера з певною ймовірністю.

При роботі з кластеризацією виникає проблема з визначенням кількості кластерів. Саме цей параметр є входним для багатьох алгоритмів. У залежності від обраної кількості кластерів розбиття на кластери може кардинально відрізнятись.

Саме тому було розроблено методи знаходження оптимальної кількості кластерів [13]. Ці методи засновані на:

- індексах, що побудовані на порівнянні ступенів «розкиду» даних всередині кластеру та між кластерами;
- функціях стійкості, що показують відповідність призначених кластерів для окремих елементів;
- статистиках, що побудовані на визначенні найбільш імовірного рішення;
- оцінюванні цільності розподілів.

Найбільш широко використовуються алгоритми на основі індексів. Розглянемо детальніше кожен.

– Calinski-Harabasz: цей алгоритм працює таким чином, що обирає кількість кластерів у якості аргументу та максимізує функцію $CH(K)$ [13]:

$$CH(K) = \frac{B(K)/(K-1)}{W(K)/(n-K)}, \text{ де} \quad (3.1)$$

$B(K)$ – зовнішня сума квадратів елементів даних з K кластерами;

$W(K)$ – внутрішня суми квадратів елементів даних з K кластерами.

Підсумок: цей метод є ефективним при роботі з даними невеликої розмірності.

– Krzanowski-Lai: метою цього алгоритму є максимізація функції $KL(K)$:

$$KL(K) = \left| \frac{Diff(K)}{Diff(K+1)} \right|, \text{ де} \quad (3.2)$$

$$Diff(K) = (K-1)^{\frac{2}{p}}W(K-1) - (K)^{\frac{2}{p}}W(K), \text{ де} \quad (3.3)$$

p – розмірність;

$W(K)$ – внутрішня сума квадратів елементів даних з K кластерами.

Підсумок: цей метод вимірює порядок мінливості внутрішніх дисперсій.

– Hartigan: цей алгоритм пропонує обрати таке мінімальне значення K , що $H(K)$ менше або дорівнює 10:

$$H(K) = (n - K - 1)(W(K)W(K+1) - 1). \quad (3.1)$$

– Kaufman and Rousseeuw: цей алгоритм пропонує вимірювати наскільки i -та точка була кластеризована. Для цього використовується наступна функція:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}, \text{ де} \quad (3.1)$$

$a(i)$ - середня відстань між i -тою точкою та всіма іншими, що потрапили в той же кластер;

$b(i)$ - середня відстань до точок у найближчому кластері, де під найближчим кластером розуміється той, який мінімізує $b(i)$.

Кількість кластерів вважається правильною, якщо вона максимізує середнє значення $s(i)$.

Розглянемо найбільш поширені алгоритми кластеризації даних, а саме: k-means [19], g-means [20] та c-means [20].

– Алгоритм k-means

Алгоритм працює таким чином, що повинен зробити мінімальним сумарне квадратичне відхилення точок кластерів від їх центрів:

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2, \text{ де} \quad (3.1)$$

k – це кількість кластерів;

x_j – це вектори об'єкта кластеризації;

S_i – це отримані кластери, $i = 1, 2, \dots, k$;

μ_i – центри мас векторів $x_j \in S_i$.

Однак у цього алгоритму є ряд недоліків:

- 1) при розв'язанні не гарантується знаходження глобального мінімуму сумарного квадратичного відхилення V ;
- 2) необхідно знати завчасно число кластерів;
- 3) результат залежить від того, якими було обрано початкові кластери. Оптимальний їх вибір невідомий.

– Алгоритм g-means

Алгоритм g-means працює таким чином, що повинен визначити кількість кластерів. Це здійснюється шляхом виконання статичного тесту. Суть статичного тесту заключається в тому, що дані в середині кожного кластера підкоряються певному унімодальному закону розподілу.

У разі негативного результату тесту кластер розбивається на два нові. Центри нових розташовані на осі головних компонент. Приклад такого розбиття наведено на рисунку 3.2.

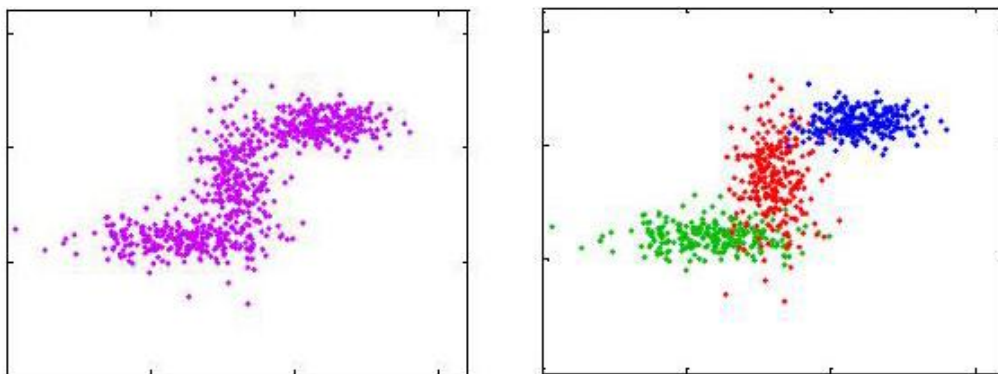


Рисунок 3.2 – Розбиття кластера на два нові

З рисунка 3.2 бачимо, що вихідну вибіку (зліва) було розбито на кластери, що підкоряються симетричному унімодальному розподілу.

Суть алгоритму g-means наступна. На початку роботи береться мале число кластерів s . У випадку, коли розподіл даних всередині кластеру не відповідає вже обраному унімодальному симетричному закону розподілу, на кожній ітерації відбувається розбиття центра одного кластера на два.

При цьому між ітераціями працює алгоритм кластеризації методом k-means. Це робиться для того, щоб максимізувати очікування для визначення нових центрів та розподілу в них даних.

Математично це виглядає так:

$$C \leftarrow \text{kmeans}(C, X) \quad (3.2)$$

У результаті розбиття центру кластера на два нові відбувається переміщення їх на першу головну компоненту розподілу відповідно до центру кластера, що розбивається.

– Алгоритм c-means

Алгоритм c-means відноситься до нечітких алгоритмів кластеризації. Основною відмінністю цього алгоритму від вище зазначених є те, що він визначає, з якою імовірністю об'єкт буде належати до певного кластеру.

Розглянемо рисунок 3.2, на якому наведено приклад розбиття на кластери за допомогою алгоритму c-means.

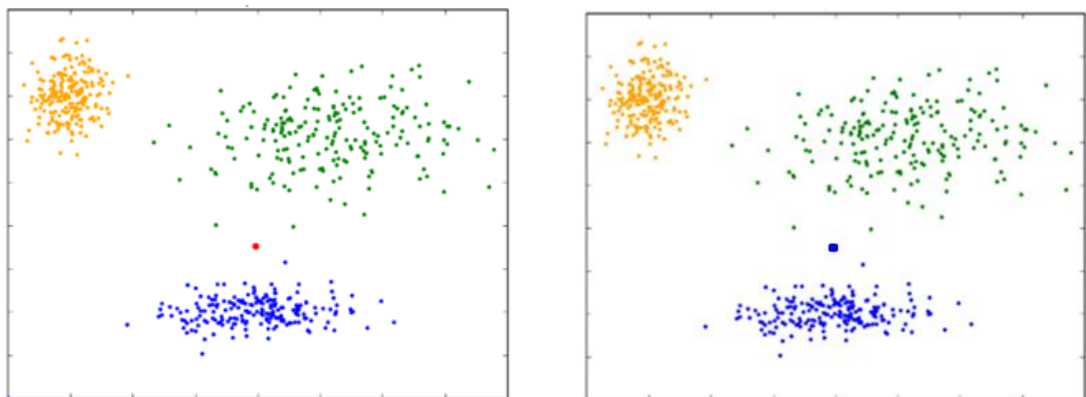


Рисунок 3.3 – Розбиття кластерів за допомогою алгоритму c-means

На рисунку 3.3 (зображення ліворуч) червоним виділено точку, яка має сумнівну позицію. Її можна віднести як до «зеленого» кластеру, так і до «синього». Але відомо, що імовірність потрапляння цієї точки в «синю» область – 70%, а в «зелену» – 30%. У результаті алгоритм c-means, використовуючи імовірність потрапляння точки в певний кластер, відніс точку до «синього» кластеру.

3.4 Опис методів розв’язання

У даному дипломному проєкті було використано алгоритм неїєрархічної кластеризації k-means. Для цього алгоритму необхідно ввести початкове значення кластерів.

– Алгоритм для знаходження кількості кластерів

Наведемо детальний опис алгоритму для знаходження кількості кластерів та алгоритму c-means.

КРОК 1. Обираємо центр ваги випадковим чином серед усіх точок.

КРОК 2. Знаходимо значення квадрата відстані до найближчого центроїда для кожної точки з тих, що вже було обрано (dx^2).

КРОК 3. Обираємо з цих точок наступний центроїд. Це необхідно робити так, щоб імовірність вибору точки була пропорційна обчисленому для неї квадрату відстані.

КРОК 3.1. Обчислюємо суму $\text{Sum}(dx^2)$.

КРОК 3.2. Обчислюємо $\text{Rnd} = \text{random}(0.0, 1.0) * \text{Sum}(dx^2)$.

КРОК 3.3. Визначаємо, якій точці з інтервалу $[0; \text{Sum})$ відповідає значення Rnd. Для цього повторюємо КРОК 3.1.

ЯКЩО сума перевищила Rnd, то КРОК 3.4.

КРОК 3.4. Беремо поточну точку в якості центроїду.

КРОК 4. Повторюємо КРОК 2 та КРОК 3 до тих під, доки не буде знайдено всі необхідні центроїди.

КРОК 5. Підрахунок кількості центроїдів.

– Алгоритм k-means

Опишемо алгоритм k-means, що було використано в даному дипломному проекті.

КРОК 1. Задаємо, що $j := 0$.

КРОК 2. Випадковим чином ставимо у відповідність кожен елемент початкової множини одному з m кластерів.

КРОК 3. Визначаємо центр кожного кластера. Для нашої задачі це елемент, компоненти якого обчислюються шляхом знаходження середнього арифметичного тих компонент, які входять у цей кластер. Після цього в центрі кластеру досягається мінімум функції суми квадратів відстаней від елементів кластера до точки.

КРОК 4. Обчислюємо відстань від кожного елементу до центру кожного кластеру. Той елементу, для якого ця відстань виявилася мінімальною, приписується до кластеру.

ЯКЩО сума квадратів відстаней від кожного елемента до центру його кластера менша за критичне значення, ТО КРОК 5.

КРОК 5. Вихід.

ІНАКШЕ КРОК 6.

КРОК 6. $j++$ та переходимо до КРОКУ 2.

Висновок до розділу

У розділі математичного забезпечення було сформульовано змістовну та математичну постановку задачі.

Задачею кластеризації є розподілення об'єктів на «схожі» групи. Такі групи називають кластерами.

Для розв'язання поставленої задачі було знайдено, описано та проаналізовано методи розв'язання.

При роботі з кластеризацією виникає проблема з визначенням кількості кластерів. Саме цей параметр є вхідним для багатьох алгоритмів, адже в залежності від обраної кількості кластерів розбиття на кластери може кардинально відрізнятись. Було визначено методи, за допомогою яких можна знайти кількість кластерів, а саме: Calinski-Harabasz, Krzanowski-Lai, Hartigan, Kaufman and Rousseeuw.

Для реалізації поставленої задачі кластеризації даних місцевих фінансів було виділено дві основні класифікації алгоритмів кластеризації: ієрархічні та неієрархічні, чіткі та нечіткі. Детально було проаналізовано алгоритми k-means, g-means, c-means. У даному дипломному проекті було вирішено використовувати алгоритм k-means. Алгоритм працює таким чином, що повинен зробити мінімальним сумарне квадратичне відхилення точок кластерів від їх центрів.

У підрозділі опису методів розв'язання було покроково розписано роботу алгоритму знаходження кількості кластерів та алгоритму кластеризації k-means.

4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

4.1 Засоби розробки

При розробці програмного продукту було використано такі засоби розробки як HTML5, CSS3, JS та MongoDB. Розглянемо детальніше ці засоби для програмування окремо для клієнтської та серверної частини.

– Клієнтська частина:

1) HTML5

HyperText Markup Language, або мова гіпертекстової розмітки сторінок, з'явилася в 1992 році та використовується дотепер [8], але, звісно, у більш вдосконаленому вигляді.

HTML – мова розмітки, що має наступні можливості:

- а) представляти документи в мережі у вигляді електронних документів із вмістом у вигляді форматowanego тексту, таблиць, списків, фото;
- б) включати в документи звукові фрагменти, а саме: відео, електронні таблиці, елементи мультимедіа;
- с) завантажувати документи шляхом активізації гіперпосилання;
- д) розробляти форми для безпосередньої роботи з віддаленими службами (пошуковими роботами, онлайн магазинами).

HTML має свою особливу розмітку. Вона заключається в тому, щоб представити документ у вигляді послідовності елементів [Кириченко А.В., Хрусталеv А.А HTML5+CSS3. Основы современного web-дизайна. – СПб.: «Наука и Техника, 2018. – 352 с.].

Документ у форматі .html складається з трьох основних частин:

- а) рядки, що оголошують файл як документ, написаний на HTML5;
- б) заголовок, що міститься в тезі HEAD;
- с) тіло документа, що представлено тегом BODY.

HTML5 – це удосконалена версія стандарту HTML, робота над якою розпочалася WHATWG у червні 2004 року та специфікація якої отримала назву «Web Application 1.0» [17].

HTML5 забезпечує скорочення використання RIA-технологій, заснованих на плагінах, а саме: Adobe Flash, Microsoft Silverlight та Sun JavaFX [Krill, Paul HTML5: Could it kill Flash and Silverlight. – InfoWorld, 2009]. Цю мову підтримують такі браузерери як Mozilla Firefox, Opera, Apple Safari та Google Chrome. Internet Explorer на даний момент не забезпечує коректне відображення сайту, написаного на HTML5.

2) CSS3

Cascading Style Sheets, або каскадні таблиці стилів, використовуються для опису зовнішнього вигляду сторінок, що написані мовами розмітки, наприклад, HTML чи XHTML. Специфікації CSS було створено Консорціумом Всесвітньої мережі. CSS має різні рівні та профілі. Рівні: CSS1, CSS2, CSS3; профілі: для принтерів, мобільних пристроїв. Профілі – це сукупність правил CSS одного або більше рівнів, що було винайдено для окремих інтерфесів [14].

Завдяки CSS один і той самий HTML документ може бути представлений у різному вигляді, залежно від прописаних стилей. Стилі поділяють на декілька категорій:

а) авторські: зовнішні та внутрішні таблиці стилів, стилі окремих елементів;

б) коистувацькі: локальний файл із розширенням .css, який користувач вказав на сторінках та в налаштуваннях браузера;

в) стилі переглядача: наприклад, стандартний стиль для елементів, що були визначені браузером. Цю категорію використовують у тому випадку, коли інформація про стиль елемента відсутня.

Основними перевагами css є:

а) інформація про стилі всіх елементів сайту може міститися в одному файлі;

б) для різних ситуацій можна відображати різні стилі. Наприклад, користувач має змогу вказати на сайті, що він має поганий зір, тоді шрифти збільшаться. Також при роботі на різних пристроях розширення екранів також різні, що за допомогою стилів CSS також можна контролювати та підбирати правильне розширення;

в) сторінки мають значно менший розмір та стають більш структурованими. Завдяки цьому швидше завантажуються, що дає менше навантаження на сервер та канали передачі. Це пов'язано з тим, що браузері мають властивість кешувати інформацію про стилі та використовувати для всіх сторінок.

CSS3 – це останній рівень мови Cascading Style Sheets. Для цього рівня було внесено такі зміни як заокруглені краї, тіні, градієнти, переходи та анімація, а також нові макети, такі як multi-columns, «резиновий» дизайн та сітковий макет [14].

– Серверна частина:

1) JS

Java Script – це динамічна, об'єктно-орієнтована [15], прототипна мова програмування [18].

Найчастіше цю мову використовують для сторони клієнта, щоб писати сценарії переходів між веб-сторінками, керувати браузером, асинхронно обмінюватися даними з сервером, а також змінювати зовнішній вигляд веб-сторінок та їх структуру.

Усе ж цю мову також використовують для:

а) односторінкових застосунків (AngularJS, React);

б) програмування на стороні сервера (як реалізовано в нашому проекті);

в) стаціонарних та мобільних застосунків;

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

- г) сценаріїв у прикладному ПЗ (Adobe Creative Suite, Apache JMeter);
- д) PDF-документів.

Синтаксис мови JS походить від C, але семантика та дизайн JS відрізняється. Це пов'язано із впливом на неї Self та Scheme.

Починаючи з 2015 року щорічно виходять нові версії JS. Наразі актуальною версією є ES2017 [16], що використана в даному дипломному проекті.

JS має анонімні функції, функції як об'єкти першого класу, автоматичне приведення типів, об'єкти з динамічною зміною типу через механізм прототипів, обробку винятків. Також JS має вбудовані об'єкти, такі як Global, Object, Error, Function, Array, String, Boolean, Number, Math, Date, RegExp. Також у цій мові є набір вбудованих операцій та операторів.

2) MongoDB

MongoDB – це документоорієнтована БД, що відноситься до нереляційних БД. Вона призначена для глибокої, масштабованої та швидкої роботи навіть для великих об'ємів даних. При проектуванні БД ще на початковому етапі закладається висока доступність, підтримка складних систем та просте розподілення даних по декількох серверах [10].

Нереляційні БД, або No SQL – це бази даних, що мають механізм видобування та зберігання даних не такий, як підхід таблиць-відношень у реляційних БД.

MongoDB зберігає дані у документах формату .json. Спеціальні запити, індексація та агрегація забезпечують потужні способи доступу до даних та їх аналізу в режимі реального часу.

Основними можливостями MongoDB є: документо-орієнтоване сховище, гнучка мова для формування запитів, повна підтримка індексів, профілювання запитів, швидкі оновлення, динамічні запити, підтримка відмовостійкості, масштабованість, ефективне зберігання бінарних даних (фото, відео).

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

MongoDB є досить молодого, але код релізу стабільний, перевіряється протягом півтора року. Її рекомендується розгортати мінімум на двох серверах використовуючи реплікацію Master/Slave [21]. Master/Slave – це модель, у якій один пристрій є головним (ведучим). Він здійснює односпрямоване управління підлеглим (веденим) пристроєм чи процесом [4].

4.2 Вимоги до технічного забезпечення

Для роботи з програмою користувачу достатньо мати комп'ютер з будь-якими технічними характеристиками. Обов'язковою умовою є встановлений на ньому веб-браузер, крім Internet Explorer, та доступ в інтернет.

У разі, коли з програмним продуктом взаємодіє розробник, то до складу технічних засобів повинні входити:

– комп'ютер із наступною конфігурацією:

- 1) процесор з тактовою частотою не нижче 1.5 ГГц;
- 2) 32- або 64-розрядна операційна система;
- 3) достатній об'єм оперативної пам'яті (не менше 2 Гб);
- 4) інші складові можуть мати будь-які параметри, тому що вони

не значним чином впливають на роботу програми;

– додатково має бути встановлене таке програмне забезпечення:

- 1) база даних MongoDB;
- 2) Studio 3T;
- 3) Sublime Text, але достатньо будь-якого текстового редактору;

– комп'ютерна периферія, до складу якої входить:

- 1) монітор;
- 2) мишка;
- 3) клавіатура.

4.3 Архітектура програмного забезпечення

4.3.1 Діаграма послідовності

Наш програмний продукт розподілено на три незалежні процеси.

Перший процес – завантаження даних по обраному місцевому бюджету.

Другий процес – візуалізація даних залежно від обраного параметру візуалізації.

Третій процес – кластеризація місцевих бюджетів.

Для цих трьох процесів на рисунках 4.1 – 4.3 наведено діаграми послідовності.

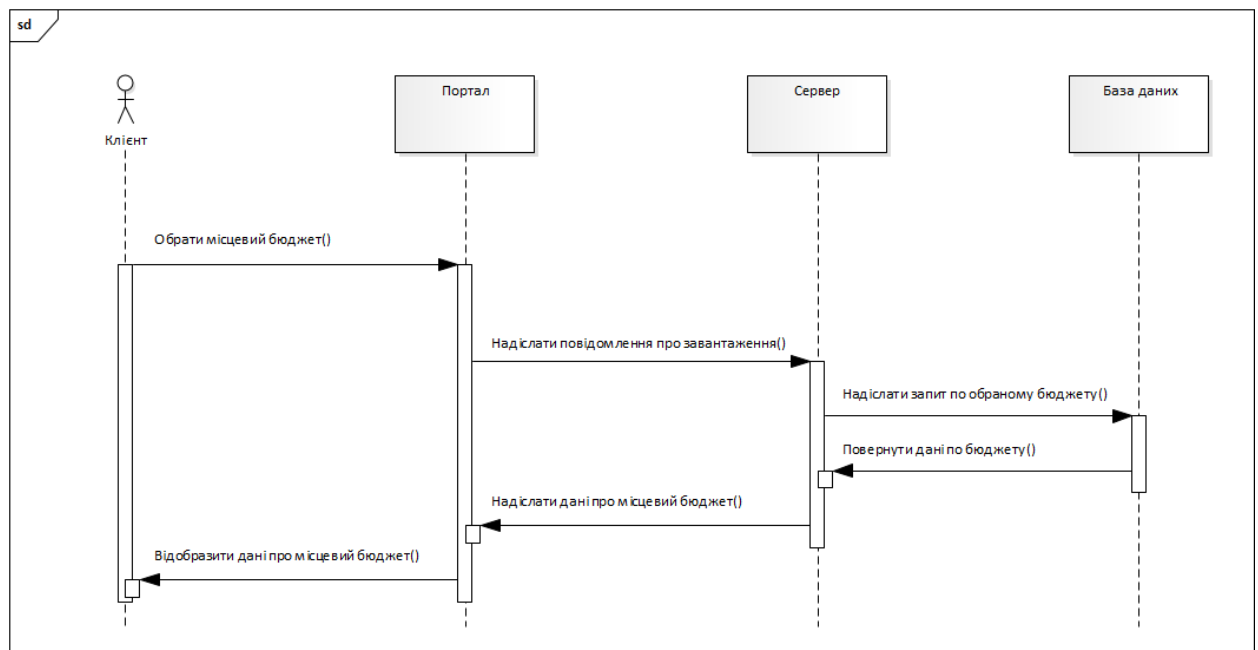


Рисунок 4.1 – Структурна схема послідовності для завантаження даних по обраному місцевому бюджету

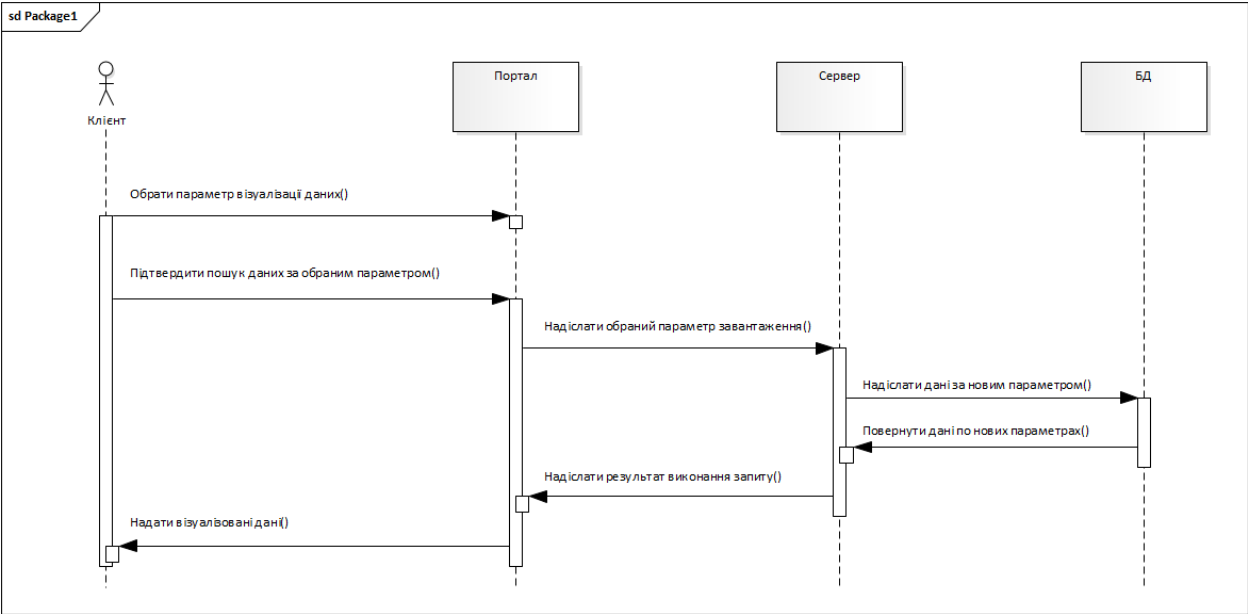


Рисунок 4.2 – Структурна схема послідовності для візуалізації даних залежно від обраного параметру візуалізації

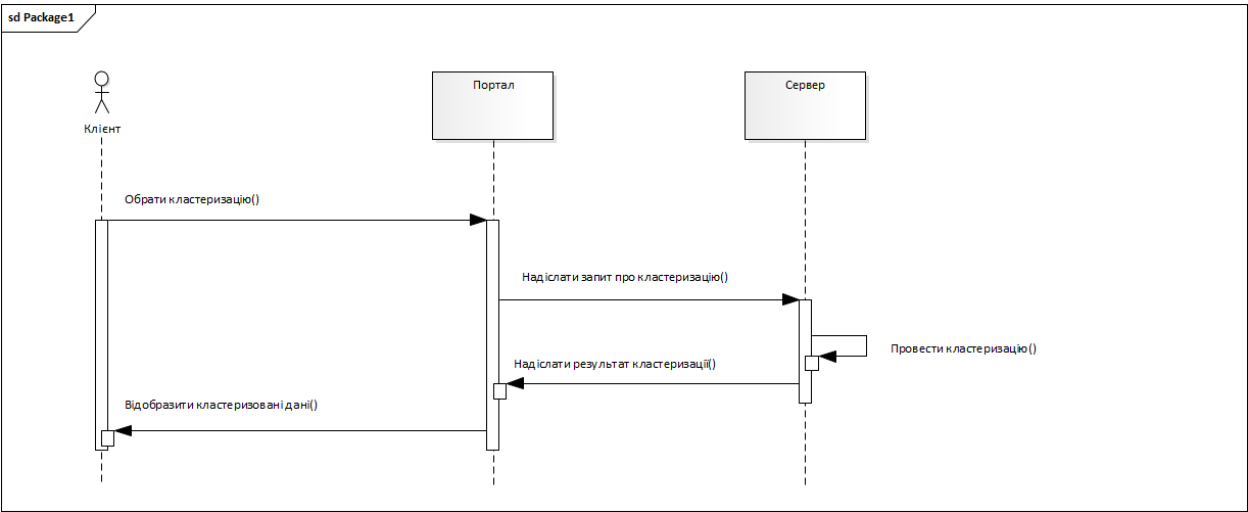


Рисунок 4.3 – Структурна схема послідовності для кластеризація місцевих бюджетів

4.3.2 Діаграма розгортання

На рисунку 4.4 наведено схему структурну розгортання.

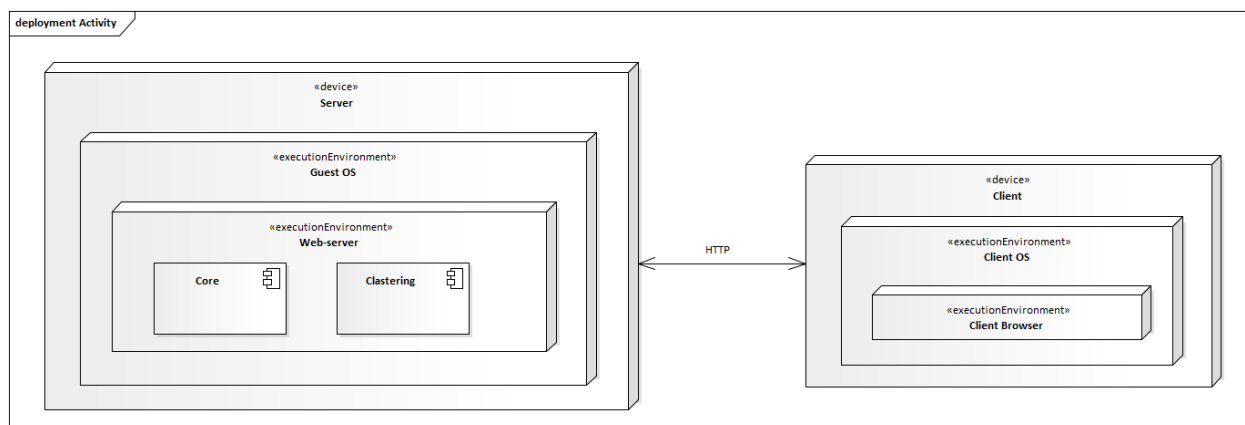


Рисунок 4.4 – Схема структурна розгортання

Опишемо більш детально схему структурну розгортання, що зображена на рисунку 4.4. Бачимо, що програма складається із серверної та клієнтської частин. Ієрархія наступна: Server, у який входить Guest OS, у який, у свою чергу, входить Web-server. Web-server складається з двох компонентів: Core та Clustering; Client, у який входить Client OS, у який, у свою чергу, входить Client Browser.

4.3.1 Специфікація функцій

Кожна функція прив'язана до певного файлу. Розглянемо окремо кожен файл та функції, які в ньому використовуються. У таблиці 4.1 наведено детальний опис функцій, що використані у файлі UkraineMap.

Таблиця 4.1 – Опис функцій із файлу UkraineMap

Назва функції	Опис
componentDidMount_ UkraineMap	Функція, що викликається при монтуванні додатку. Для нашого застосунку початково обраний регіон – це Київ)
componentWillReceiveProps_ UkraineMap (nextProps)	Функція, що виділяє нову область на карті при виборі нової області в Select
highlightRegion_ UkraineMap (event)	Функція, що активує область на карті при натисненні на неї. Якщо область вже була активною, то ніяких дій не відбувається

У таблиці 4.2 наведено детальний опис функцій, що використані у файлі MainPage.

Таблиця 4.2 – Опис функцій із файлу MainPage

Назва функції	Опис
changeRegionFromMap_ MainPage	Функція, що змінює регіон в Select при зміні регіону на карті
changeRegion_ MainPage	Функція, що передає дані в разі зміни регіону вручну.
changeCity_ MainPage	Якщо користувач вручну змінює регіон, то функція змінює міста в залежності від обраного регіону
chooseCities_ MainPage	Функція, що формує список усіх регіонів
chooseRegion_ MainPage	Функція, що формує список усіх міст у залежності від регіону
render_ MainPage	Функція, що візуалізує дані на сайті
componentDidMount_ MainPage	Функція для встановлення початкових даних

У таблиці 4.2 наведено детальний опис функцій, що використані у файлі Income.

Таблиця 4.2 – Опис функцій із файлу Income

Назва функції	Опис
handleClick_ Income	Функція, що повідомляє системі, яку діаграму бажає переглянути користувач.
handleSearch_ Income	При натисненні на кнопку «Пошук», то функція бере нові дані з БД
resetFlags_ Income	Функція, що викликається при натисненні на кнопку «Доходи»
render_ Income	Функція, що візуалізує дані на сайті
componentDidMount_ Income	Функція для встановлення початкових даних

У таблиці 4.3 наведено детальний опис функцій, що використані у файлі ChartExplanation.

Таблиця 4.3 – Опис функцій із файлу ChartExplanation

Назва функції	Опис
updateChart ChartExplanation	Функція, що створює діаграму з даними, що прийшли від handleClick_ Income
render_ ChartExplanation	Функція, що візуалізує дані на сайті
componentDidMount_ ChartExplanation	Функція для встановлення початкових даних

У таблиці 4.4 наведено детальний опис функцій, що використані у файлі ChartStructure.

Таблиця 4.4 – Опис функцій із файлу ChartStructure

Назва функції	Опис
updateChart ChartStructure	Функція, що створює діаграму з даними, що прийшли від handleClick_ Income
render_ ChartStructure	Функція, що візуалізує дані на сайті
componentDidMount_ ChartStructure	Функція для встановлення початкових даних

У таблиці 4.5 наведено детальний опис функцій, що використані у файлі ChartDynamics.

Таблиця 4.5 – Опис функцій із файлу ChartDynamics

Назва функції	Опис
updateChart ChartDynamics	Функція, що створює діаграму з даними, що прийшли від handleClick_ Income
render_ ChartDynamics	Функція, що візуалізує дані на сайті
componentDidMount_ ChartDynamics	Функція для встановлення початкових даних

У таблиці 4.6 наведено детальний опис функцій, що використані у файлі Explanation.

Таблиця 4.6 – Опис функцій із файлу Expenditures

Назва функції	Опис
handleClick _ Expenditures	Функція, що повідомляє системі, яку діаграму бажає переглянути користувач.
handleSearch_ Expenditures	При натисненні на кнопку «Пошук», то функція бере нові дані з БД
resetFlags_ Expenditures	Функція, що викликається при натисненні на кнопку «Видатки»
render_ Expenditures	Функція, що візуалізує дані на сайті
componentDidMount_ Expenditures	Функція для встановлення початкових даних

У таблиці 4.7 наведено детальний опис функцій, що використані у файлі Preloader.

Таблиця 4.7 – Опис функцій із файлу Preloader

Назва функції	Опис
render_ Preloader	Функція, що візуалізує Preloader
componentDidMount_ Preloader	Функція для встановлення початкових даних

У таблиці 4.8 наведено детальний опис функцій, що використані у файлі Clusterization.

Таблиця 4.8 – Опис функцій із файлу Clusterization

Назва функції	Опис
componentDidMount_ Clusterization	Функція для встановлення початкових даних
updateCluster_Clusterization(data)	Функція, що бере дані із серверу про кластеризацію, приймає ці значення та записує
render_Clusterization	Функція, що візуалізує дані на сайті

4.4 Опис звітів

У даному дипломному проекті звіти представлено у вигляді візуалізованих даних:

- візуалізовані дані щодо виконання бюджету по доходах;

- візуалізовані дані щодо виконання бюджету по витратам;
- візуалізовані дані щодо структури бюджету по доходах;
- візуалізовані дані щодо структури бюджету по витратах;
- візуалізовані дані щодо динаміки бюджету, представлені помісячно.

На рисунку 4.5 наведено діаграму, на якій зображено дані по виконанню місцевих бюджетів по доходах за 2018 рік до уточненого річного плану. Видно, що до уточненого річного плану було виконано 104,2%.



Рисунок 4.5 – Діаграма виконання бюджету по доходах

На рисунку 4.6 наведено діаграму, на якій зображено дані по виконанню місцевих бюджетів по видатках за 2018 рік до уточненого річного плану. Видно, що до уточненого річного плану було виконано 100,7%.



Рисунок 4.6 – Діаграма виконання бюджету по видаткам

На рисунку 4.7 наведено діаграму, на якій зображено дані по структурі бюджету по доходах за 2018 рік. Бачимо, що бюджетна система по доходах складається з таких показників: податкові надходження (63,86%), трансфери

з державного бюджету (26,95%), неподаткові надходження (8,24%), цільові фонди (0,53%), доходи від операцій з капіталом (0,42%).



Рисунок 4.7 – Діаграма структури бюджету по доходах

На рисунку 4.8 наведено діаграму, на якій зображено дані по структурі бюджету по видатках за 2018 рік. Бачимо, що бюджетна система по видатках складається з таких показників: економічна діяльність (28,21%), освіта (25,72%), охорона здоров'я (17,61%), соціальний захист та соціальне забезпечення (14,98%), житлово-комунальне господарство (6,27%), загальнодержавні функції (3,82%), духовний та фізичний розвиток (2,80%), міжбюджетні трансфери (0,27%), охорона навколишнього природного середовища (0,21%), громадський порядок, безпека та судова влада (0,11%).

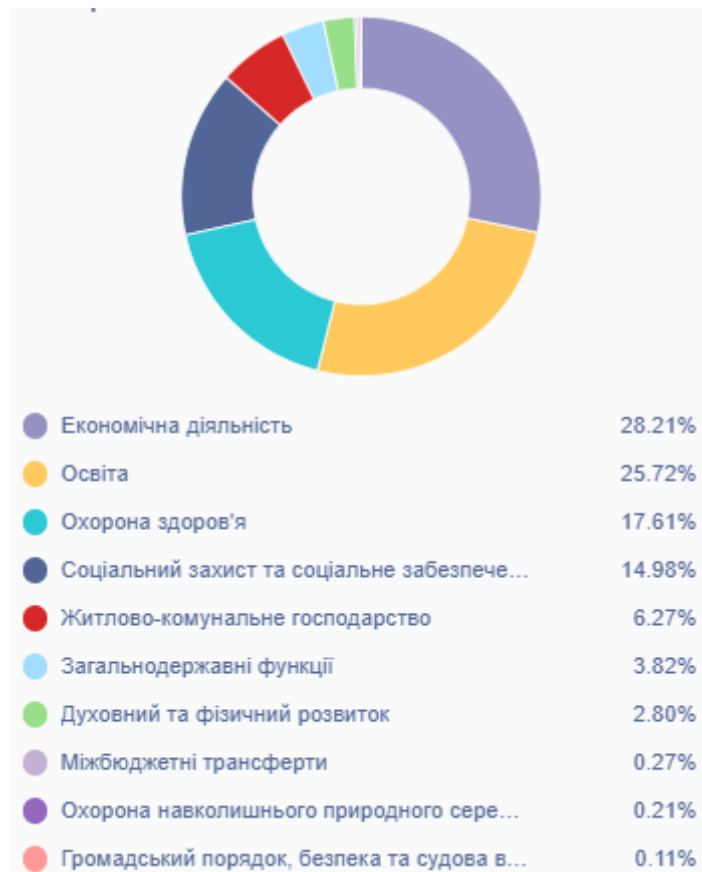


Рисунок 4.8 – Діаграма структури бюджету по видатках

На рисунку 4.9 наведено діаграму, на якій зображено динаміку бюджету помісячно. Горизонтальна вісь – це дані по місяцям від січня до грудня, вертикальна вісь – це суми, записані в мільйонах грн. Для кожного місяця точкою відмічено суму доходів та видатків. Точки для кожного місяця з'єднані між собою, що дає представлення про динаміку місцевих бюджетів.

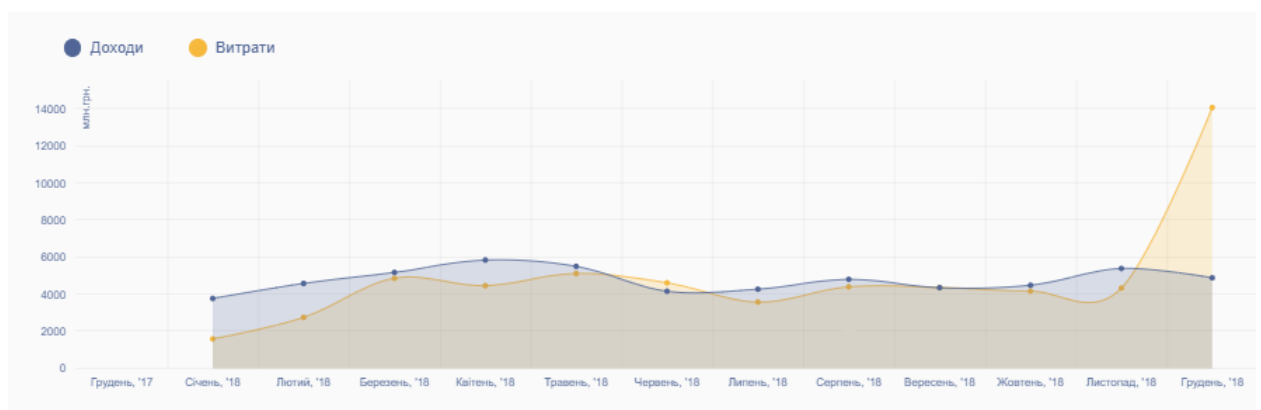


Рисунок 4.9 – Діаграма динаміки бюджету

Висновок до розділу

У розділі програмного та технічного забезпечення було описано засоби розробки, що використовувалися для створення застосунку для дипломного проекту. При розробці програмного продукту було використано такі засоби розробки як HTML5, CSS3, JS та MongoDB.

У підрозділі з вимогами до технічного забезпечення було визначено, що для роботи з програмою користувачу достатньо мати комп'ютер з будь-якими технічними характеристиками. Обов'язковою умовою є встановлений на ньому веб-браузер, крім Internet Explorer, та доступ в інтернет. Також було описано додаткові технічні характеристики при необхідності роботи безпосередньо розробникам.

У підрозділі з описом архітектури програмного забезпечення було наведено діаграму послідовності, діаграму розгортання та описано специфікацію функцій, що використовувалися для написання програмного продукту.

Також було описано звіти. У даному дипломному проєкті звіти представлено у вигляді візуалізованих даних:

- візуалізовані дані щодо виконання бюджету по доходах;
- візуалізовані дані щодо виконання бюджету по витратах;
- візуалізовані дані щодо структури бюджету по доходах;
- візуалізовані дані щодо структури бюджету по витратах;
- візуалізовані дані щодо динаміки бюджету, представлені помісячно.

5 ТЕХНОЛОГІЧНИЙ РОЗДІЛ

5.1 Керівництво користувача

Для роботи із системою користувачу необхідно мати мінімальні навички роботи із комп'ютером. Для зручності розуміння функціонування програмного продукту на рисунку 5.1 наведено схему переходу між сторінками.



Рисунок 5.1 – Схема переходів між сторінками проекту

Як видно з рисунку 5.1 робота з програмою починається з головної сторінки. З неї користувач може здійснити перехід на сторінку з інформацією про доходи. На цій сторінці вже є навігаційне меню, звідки можуть здійснюватися переходи між усіма іншими сторінками, крім головної.

Опишемо детальніше основні дії, що користувач може виконувати із запропонованою системою.

На рисунку 5.1 наведено Головну сторінку.

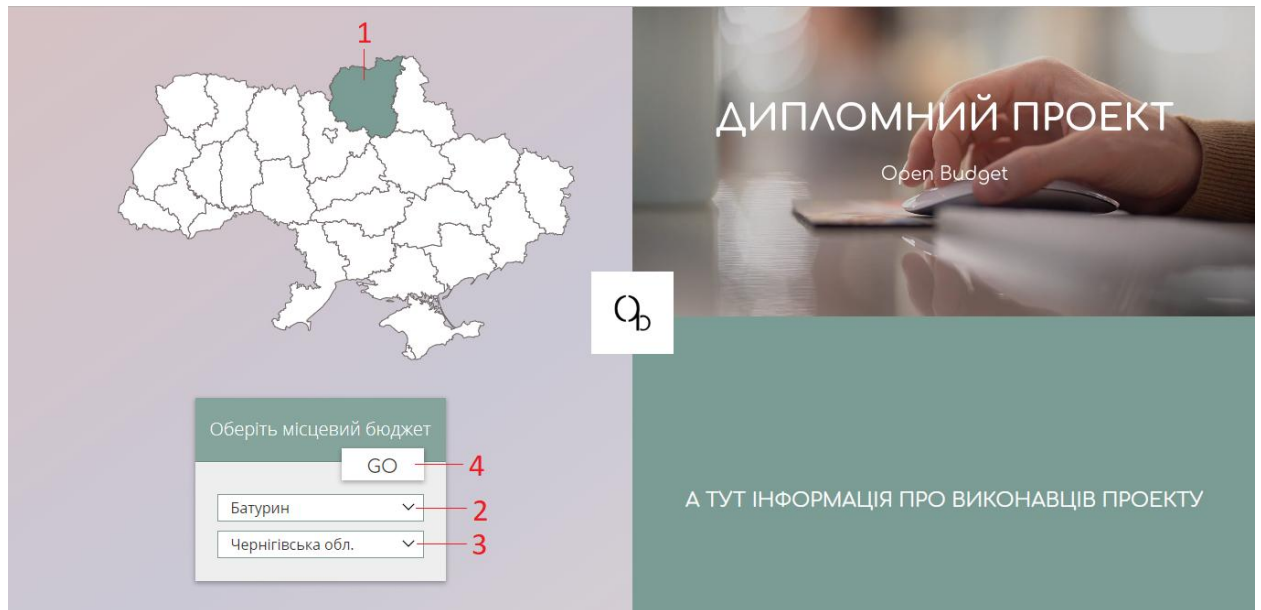


Рисунок 5.1 – Головна сторінка

Опис реакції системи на дії користувача на головній сторінці наведено в таблиці 5.1.

Таблиця 5.1 – Опис реакції системи на дії користувача при роботі з головною сторінкою

№	Опис – реакція системи
1	При наведенні на карту України підсвічується потрібна область. При натисненні на підсвічену область відбувається передача даних до елементів під номером 2 та 3.
2	При натисненні на область відкривається список із назвами населених пунктів, де користувач може обрати потрібний.
3	При натисненні на область відкривається список із переліком усіх областей, де користувач може обрати потрібну.
4	При натисненні кнопки «GO» система відкриває сторінку Доходів по обраній області та населеному пункту.

Після переходу на сторінку доходів з'являється головне меню. Розглянемо це меню окремо, оскільки воно буде використовуватися на решті сторінок.

На рисунку 5.2 наведено головне меню.

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

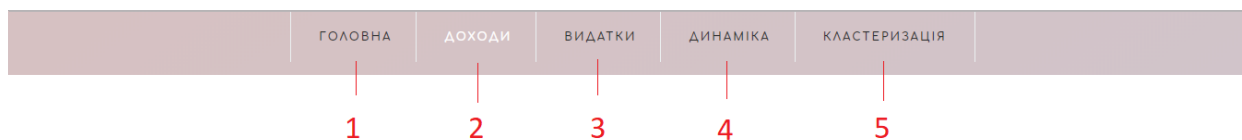


Рисунок 5.2 – Головне меню

Опис реакції системи на дії користувача на головному меню наведено в таблиці 5.2.

Таблиця 5.2 – Опис реакції системи на дії користувача при роботі з головним меню

№	Опис – реакція системи
1	При натисненні на кнопку «Головна» система відкриває головну сторінку.
2	При натисненні на кнопку «Доходи» система відкриває сторінку з інформацією про доходи.
3	При натисненні на кнопку «Видатки» система відкриває сторінку з інформацією про видатки.
4	При натисненні на кнопку «Динаміка» система відображає сторінку з візуальним представленням динаміки бюджету.
5	При натисненні на кнопку «Кластеризація» система відображає сторінку з візуальним представлення кластеризованих даних.

На рисунку 5.3 наведено сторінку з інформацією про доходи.

Код	Найменування	Розпис на рік	Виконано за період	Виконання до уточненого річного розпису, %
10000000	Податкові надходження	14036200	1492987.7	10.63
11000000	Податки на доходи, податки на прибуток, податки на збільшення ринкової вартості	7856800	533836.23	6.79
11010000	Податок та збір на доходи фізичних осіб	7856800	533836.23	6.79
11010100	Податок на доходи фізичних осіб, що сплачується податковими агентами, із доходів платника податку у вигляді заробітної плати	6453600	516368.59	8
11010200	Податок на доходи фізичних осіб з грошового забезпечення, грошових винагород та інших виплат, одержаних військовослужбовцями та особами рядового і начальницького складу, що сплачується податковими агентами	155700	14802.72	9.5
11010400	Податок на доходи фізичних осіб, що сплачується податковими агентами, із доходів платника податку інших ніж заробітна плата	1174000	193.7	0.01
11010500	Податок на доходи фізичних осіб, що сплачується фізичними особами за результатами річного декларування	73500	2471.22	3.36
13000000	Рентна плата та плата за використання інших природних ресурсів	25300	0	0
13010000	Рентна плата за спеціальне використання лісових ресурсів	25300	0	0
13010200	Рентна плата за спеціальне використання лісових ресурсів (крім рентної плати за спеціальне використання лісових ресурсів в частині деревини, заготовленої в порядку рубок головного користування)	25300	0	0

Рисунок 5.3 – Сторінка з інформацією про доходи

Опис реакції системи на дії користувача на сторінці з інформацією про доходи наведено в таблиці 5.2.

Таблиця 5.2 – Опис реакції системи на дії користувача при роботі зі сторінкою з інформацією про доходи

№	Опис – реакція системи
1	При натисненні на область відображається випадаючий список, де користувач має можливість обрати рік, по якому відображатиметься інформація по доходах.
2	При натисненні на область відображається випадаючий список, де користувач має можливість обрати початковий місяць періоду, за який відображатимуться дані по доходах.
3	При натисненні на область відображається випадаючий список, де користувач має можливість обрати кінцевий місяць періоду, за який відображатимуться дані по доходах.
4	При натисненні на область відображається випадаючий список, де користувач має можливість обрати код, за яким відображатимуться дані по доходах.
5	При натисненні на область система відображає дані по доходах в таблиці за обраними параметрами в областях 1, 2, 3, 4.
6	При натисненні на кнопку «Виконання бюджету» система відкриває вікно, де візуалізовано дані щодо виконання бюджету по доходах (рисунок 5.4).
7	При натисненні на кнопку «Структура бюджету» система відкриває вікно, де візуалізовано дані щодо структури бюджету по доходах (рисунок 5.5).

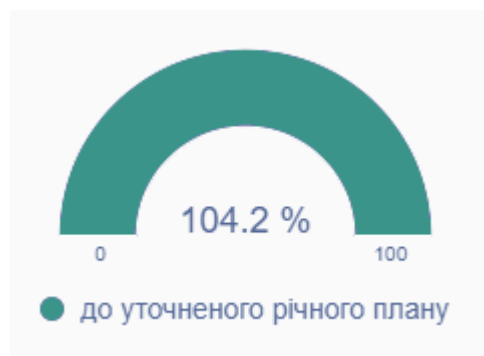


Рисунок 5.4 – Вікно з інформацією про виконання бюджету



Рисунок 5.5 – Вікно з інформацією про структуру бюджету

На рисунку 5.6 наведено сторінку з інформацією про видатки.



Рисунок 5.6 – Сторінка з інформацією про видатки

Опис реакції системи на дії користувача на сторінці з інформацією про видатки наведено в таблиці 5.3.

Таблиця 5.3 – Опис реакції системи на дії користувача при роботі зі сторінкою з інформацією про видатки

№	Опис – реакція системи
1	При натисненні на область відображається випадаючий список, де користувач має можливість обрати рік, по якому відображатиметься інформація по видатках.
2	При натисненні на область відображається випадаючий список, де користувач має можливість обрати початковий місяць періоду, за який відображатимуться дані по видатках.
3	При натисненні на область відображається випадаючий список, де користувач має можливість обрати кінцевий місяць періоду, за який відображатимуться дані по видатках.
4	При натисненні на область відображається випадаючий список, де користувач має можливість обрати код, за яким відображатимуться дані по видатках.
5	При натисненні на область система відображає дані по видатках в таблиці за обраними параметрами в областях 1, 2, 3, 4.
6	При натисненні на кнопку «Виконання бюджету» система відкриває вікно, де візуалізовано дані щодо виконання бюджету по видатках.
7	При натисненні на кнопку «Структура бюджету» система відкриває вікно, де візуалізовано дані щодо структури бюджету по видатках.

На рисунку 5.7 наведено сторінку з інформацією про динаміку.

ГОЛОВНА	ДОХОДИ	ВИДАТКИ	ДИНАМІКА	КЛАСТЕРИЗАЦІЯ
---------	--------	---------	----------	---------------

Динаміка бюджету (помісячно)

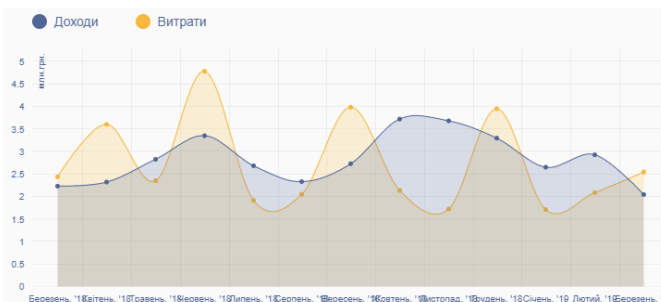


Рисунок 5.7 – Сторінка з інформацією про динаміку

При наведенні на графік система відображає детальну інформацію про доходи та витрати за кожен місяць.

На рисунку 5.8 наведено сторінку з інформацією про кластеризацію.

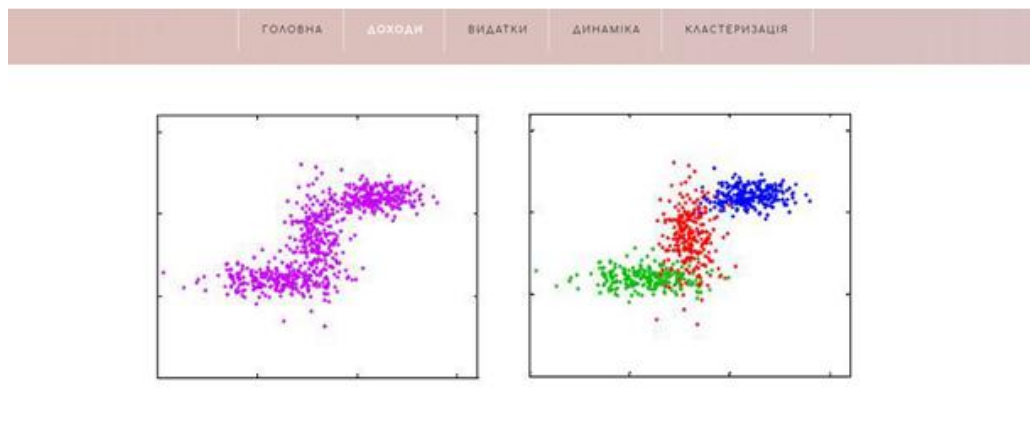


Рисунок 5.8 – Сторінка з інформацією про кластеризацію

5.2 Випробування програмного продукту

5.2.1 Мета випробувань

Метою випробувань являється перевірка відповідності функцій комплексу задач кластеризації даних місцевих фінансів вимогам технічного завдання.

5.2.2 Загальні положення

Випробування проводяться на основі наступних документів:

- ГОСТ 34.603–92. Інформаційна технологія. Види випробувань автоматизованих систем;
- ГОСТ РД 50-34.698-90. Автоматизовані системи вимог до змісту документів.

5.2.3 Результати випробувань

Розглянемо основні функції системи та перевіримо їх відповідність вимогам технічного завдання. У таблицях 5.4 – 5.12 наведено перелік випробувань основних функціональних можливостей.

Таблиця 5.4 – Перевірка заповнення усіх атрибутів при виборі пошуці місцевого бюджету

Початковий стан	Відкрита сторінка «Головна»
Вхідні дані	Обрана в полі «Територія» територія з наведеного списку, а в полі «Бюджет» бюджет із наведеного списку
Схема проведення тесту	Обрати територію та бюджет зі списків та натиснути кнопку «ОК»
Очікуваний результат	Відкрита сторінка «Доходи»
Стан після випробування	Відкрита сторінка «Доходи»
У разі помилки	Підсвічується червоним поле, де користувач не ввів дані, та впливає повідомлення про помилку

Таблиця 5.5 – Перевірка заповнення усіх атрибутів при пошуці місцевого бюджету по доходам

Початковий стан	Відкрита сторінка «Доходи»
Вхідні дані	Рік, обраний в полі «Місяць початку» місяць зі списку, обраний в полі «Місяць завершення» місяць зі списку, код
Схема проведення тесту	Заповнити всі поля та натиснути кнопку «Пошук»
Очікуваний результат	На сторінці «Доходи» з'являється таблиця із детальним описом місцевого бюджету по доходам та кнопка «Завантажити»
Стан після випробування	З'явилася таблиця із детальним описом місцевого бюджету по доходам та кнопка «Завантажити»
У разі помилки	Підсвічується червоним поле, де користувач не ввів дані, або ввів неправильно, та впливає повідомлення про помилку.

Таблиця 5.6 – Перевірка заповнення усіх атрибутів при пошуці місцевого бюджету по видаткам

Початковий стан	Відкрита сторінка «Видатки»
Вхідні дані	Рік, обраний в полі «Місяць початку» місяць зі списку, обраний в полі «Місяць завершення» місяць зі списку, код
Схема проведення тесту	Заповнити всі поля та натиснути кнопку «Пошук»
Очікуваний результат	На сторінці «Видатки» з'являється таблиця із детальним описом місцевого бюджету по доходам та кнопка «Завантажити»
Стан після випробування	З'явилася таблиця із детальним описом місцевого бюджету по доходам та кнопка «Завантажити»
У разі помилки	Підсвічується червоним поле, де користувач не ввів дані, або ввів неправильно, та впливає повідомлення про помилку.

Таблиця 5.7 – Перевірка можливості завантаження даних по доходам

Початковий стан	Відкрита сторінка «Доходи» із відображеною таблицею з детальним описом місцевого бюджету
Вхідні дані	Тип файлу для завантаження зі списку
Схема проведення тесту	Натиснути кнопку «Завантажити»
Очікуваний результат	На комп'ютер починається завантаження файлу
Стан після випробування	На комп'ютер починається завантаження файлу

Таблиця 5.8 – Перевірка можливості завантаження даних по видаткам

Початковий стан	Відкрита сторінка «Видатки» із відображеною таблицею з детальним описом місцевого бюджету
Вхідні дані	Тип файлу для завантаження зі списку
Схема проведення тесту	Натиснути кнопку «Завантажити»
Очікуваний результат	На комп'ютер починається завантаження файлу
Стан після випробування	На комп'ютер починається завантаження файлу

Таблиця 5.9 – Перевірка можливості здійснення перегляду структури місцевого бюджету по доходам

Початковий стан	Відкрита сторінка «Доходи» із відображеною таблицею з детальним описом місцевого бюджету
Схема проведення тесту	Натиснути кнопку «Структура бюджету»
Очікуваний результат	Відкрита сторінка «Структура бюджету по доходам»
Стан після випробування	Відкрита сторінка «Структура бюджету по доходам»

Таблиця 5.10 – Перевірка можливості здійснення перегляду структури місцевого бюджету по видаткам

Початковий стан	Відкрита сторінка «Видатки» із відображеною таблицею з детальним описом місцевого бюджету
Вхідні дані	
Схема проведення тесту	Натиснути кнопку «Структура бюджету»
Очікуваний результат	Відкрита сторінка «Структура бюджету по видаткам»
Стан після випробування	Відкрита сторінка «Структура бюджету по видаткам»

Таблиця 5.11 – Перевірка можливості здійснення перегляду динаміки бюджету помісячно

Початковий стан	Відкрита сторінка «Доходи» або сторінка «Видатки»
Схема проведення тесту	Натиснути кнопку «Динаміка бюджету»
Очікуваний результат	Відкрита сторінка «Динаміка бюджету»
Стан після випробування	Відкрита сторінка «Динаміка бюджету»

Таблиця 5.12 – Перевірка можливості здійснення перегляду кластеризації бюджету

Початковий стан	Відкрита сторінка «Доходи» або сторінка «Видатки»
Схема проведення тесту	Натиснути кнопку «Кластеризація»
Очікуваний результат	Відкрита сторінка «Кластеризація»
Стан після випробування	Відкрита сторінка «Кластеризація»

Висновок до розділу

У технологічному розділі було наведено керівництво користувача, де описано, які конкретно дії може виконувати користувач із системою. Для зручності розуміння функціонування програмного продукту було наведено схему переходу між сторінками.

У підрозділі з випробуваннями програмного продукту було описано мету випробувань, загальні положення та наведено результати випробувань.

Метою випробувань являється перевірка відповідності функцій комплексу задач кластеризації даних місцевих фінансів вимогам технічного завдання.

Випробування проводилися шляхом функціонального тестування. Було описано основні функції системи та перевірено їх відповідність вимогам технічного завдання.

Для кожної функціональної вимоги було прописано окреме випробування, у якому заповненими полями є: початковий стан, вхідні дані, схема проведення тесту, очікуваний результат, стан після випробування, результат у разі помилки.

У результаті проведення випробувань було виправлено всі виявлені помилки.

ЗАГАЛЬНІ ВИСНОВКИ

У результаті виконання дипломного проекту було розроблено програму для комплексу задач кластеризації даних місцевих фінансів.

Запропонований програмний продукт присвячений роботі з даними місцевих бюджетів.

У розділі опису програмного середовища було детально описано предметну область, процес діяльності та функціональної моделі. Опис процесу діяльності представлено у вигляді трьох незалежних процесів. Специфікацію функціональної поведінки системи представлено у вигляді діаграми варіантів використання. Було визначено, що безпосередньо із системою будуть взаємодіяти два актори: клієнт – це особа, яка може працювати із основними функціями системи; портал (зовнішній клієнт) – це наша система, яка формує веб-сторінки. Відповідно до визначених варіантів використання було виявлено функціональні вимоги та встановлено їх пріоритетність. Перед початком роботи над дипломним проектом було здійснено пошук застосунків зі схожою функціональністю. Нині існують програмні продукти, які здійснюють огляд місцевих фінансів, але для аналізу даних місцевих бюджетів ці застосунки не використовують кластеризацію, що є ключовим для даної роботи.

У розділі інформаційного забезпечення було описано вхідні та вихідні дані, структуру БД. Вхідними даними є файли з інформацією по доходах та видатках, вихідними даними є карти Кохонена, що відображають кластеризований бюджет. Для даного дипломного проекту було прийнято рішення використовувати нереляційну БД MongoDB.

У розділі математичного забезпечення було сформульовано змістовну та математичну постановку задачі, описано методи розв'язання. Було визначено методи, за допомогою яких можна знайти кількість кластерів, а саме: Calinski-Harabasz, Krzanowski-Lai, Hartigan, Kaufman and Rousseeuw. Також було описано алгоритми кластеризації k-means, g-means та c-means. У

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

даному дипломному проєкті було використано алгоритм неїєрархічної кластеризації k-means.

У розділі програмного та технічного забезпечення було описано засоби розробки HTML5, CSS3, JS та MongoDB, а також визначено вимоги до технічного забезпечення. В архітектурі програмного забезпечення було наведено діаграми послідовності та розгортання, а також описано специфікацію функцій та описано звіти.

У технологічному розділі було наведено керівництво користувача, де описано, які конкретно дії може виконувати користувач із системою. Для зручності розуміння функціонування програмного продукту було наведено схему переходу між сторінками. У підрозділі з випробуваннями програмного продукту було описано мету випробувань, загальні положення та наведено результати випробувань. Метою випробувань являється перевірка відповідності функцій комплексу задач кластеризації даних місцевих фінансів вимогам технічного завдання. Випробування проводилися шляхом функціонального тестування. Було описано основні функції системи та перевірено їх відповідність вимогам технічного завдання. Для кожної функціональної вимоги було прописано окреме випробування.

Розроблений програмний продукт є корисним для фахівців, що працюють у сфері економіки, а саме для тих, хто займається питаннями фінансування місцевих бюджетів. Для бюджетів зі «схожими» характеристиками можна підбирати однакові чи подібні програми розвитку.

У перспективі цей проєкт можна розвивати шляхом реалізації інших методів аналізу даних, а саме: класифікації, прогнозування. У комплексі з кластеризацією ці методи дадуть потужний апарат для вирішення серйозних проблем місцевих бюджетів.

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

ПЕРЕЛІК ПОСИЛАНЬ

1. А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод
Методы и модели анализа данных: OLAP и Data Mining. – СПб.: БХВ-Петербург, 2009. – 512 с.
2. Бюджетный кодекс України (від 8 липня 2010 р. № 2456-VI) (із змінами і доповненнями) // Голос України. – 2010. – № 143. – 4 серпня. / Офіційний сайт Верховної Ради України. [Електронний ресурс]. – Режим доступу: <http://zakon2.rada.gov.ua/laws/show/2456-17/page>
3. Бюджетный кодекс України, Закон від 21.06.2001 №2542-III
4. Ведущий – Ведомый. [Електронний ресурс]. – Режим доступу: [https://ru.wikipedia.org/wiki/ Ведущий_—_ведомый](https://ru.wikipedia.org/wiki/Ведущий_—_ведомый)
5. Відкритий бюджет – бета. [Електронний ресурс]. – Режим доступу: <http://openbudget.in.ua/>
6. Відкритий бюджет Кривого Рогу. [Електронний ресурс]. – Режим доступу: <https://openbudget.krmisto.gov.ua/>
7. Відкритий бюджет Херсонської області. [Електронний ресурс]. – Режим доступу: <https://openbudget.ks.ua/>
8. Дронов В.А. HTML5, CSS3 и Web 2.0. – БХВ-Петербург, 2011. – 256 с.
9. Інструкція про застосування Плану рахунків бухгалтерського обліку банків України. – Правління Національного Банку України, 2004.
10. Кайл Бенкер MongoDB в действии: навч. посібник. – ДМК Пресс, 2012. – 394 с.
11. Конституція України [Електронний ресурс] // Відомості Верховної Ради України (ВВР). – 1996. – №30. – с.141. – Режим дотупу: <https://www.president.gov.ua/ua/documents/constitution/konstituciya-ukrayini-rozdil-ix>
12. Офіційний портал Києва. [Електронний ресурс]. – Режим доступу:

					ДП ІС-5110.1153-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

https://kyivcity.gov.ua/finansy_ta_biudzheth/biudzheth/vidkrytyi_biudzheth_Shortcut_227363.html

13. Шалимов. Д.С. Алгоритмы устойчивой кластеризации на основе индексных функций и функций устойчивости. [Электронный ресурс]. – Режим доступа:<http://www.meta.math.spbu.ru/user/gran/soi4/shalymov4.pdf>

14. CSS. [Электронный ресурс]. – Режим доступа: <https://uk.wikipedia.org/wiki/CSS>

15. ECMAScript Language Specification. – Ecma International, 2011. – 246 с.

16. ES6, ES8, ES2017: что такое ECMAScript и чем это отличается от JavaScript <https://tproger.ru/translations/wtf-is-ecmascript/>

17. HTML 5. [Электронный ресурс]. – Режим доступа: https://uk.wikipedia.org/wiki/HTML#Проект_специфікації_HTML_5

18. Java Script. [Электронный ресурс]. – Режим доступа: <https://uk.wikipedia.org/wiki/JavaScript>

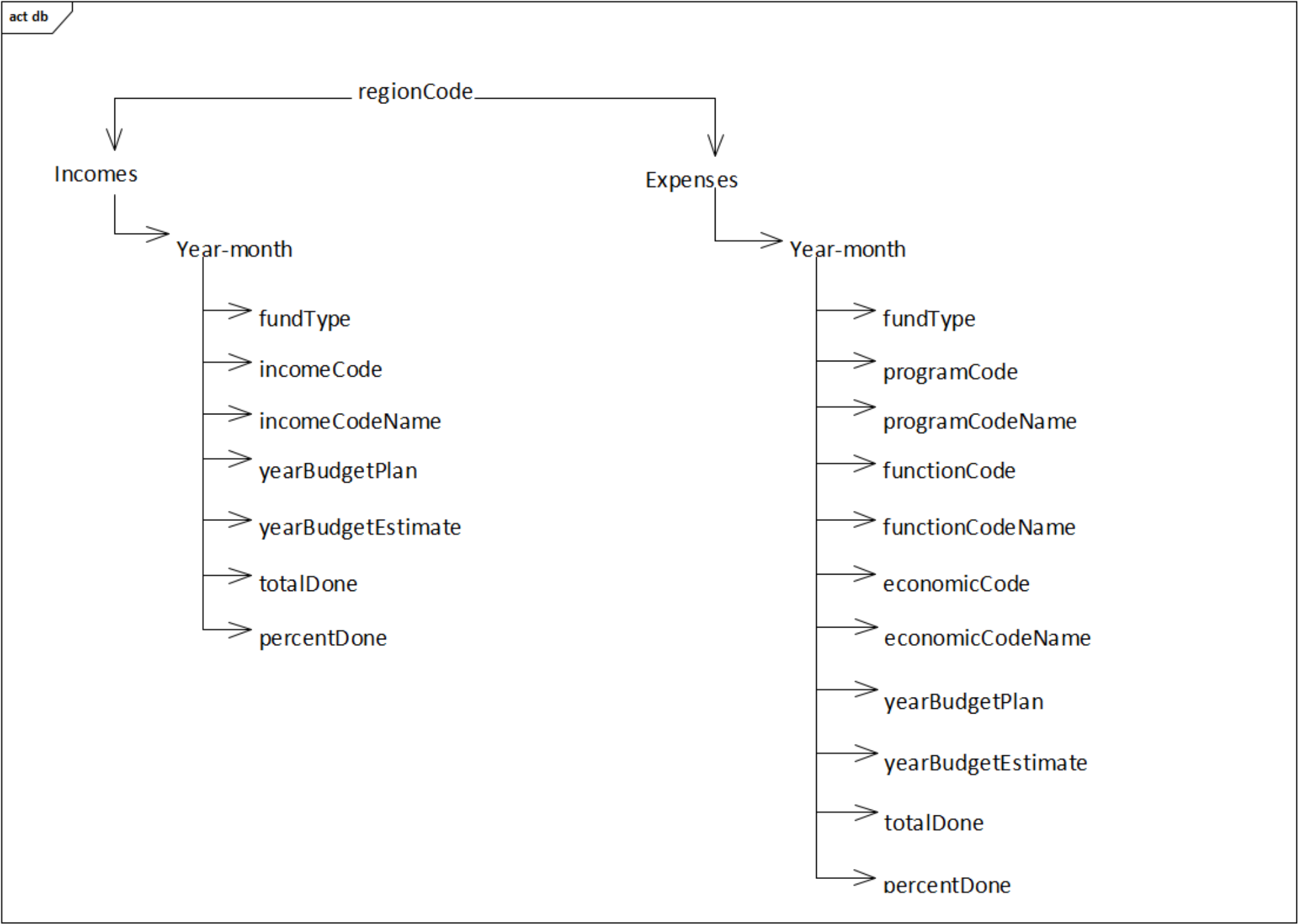
19. Learning the k in k-means, 2003, Hamerly G., Elkan C.. 3. Self-Organizing Maps, 3rd, 2001, Kohonen T.

20. Mario Koppen, Nikola Kasabov, George Coghill Advance in Neuro-Information Processing. – ICONIP, 2008. – 342 с.

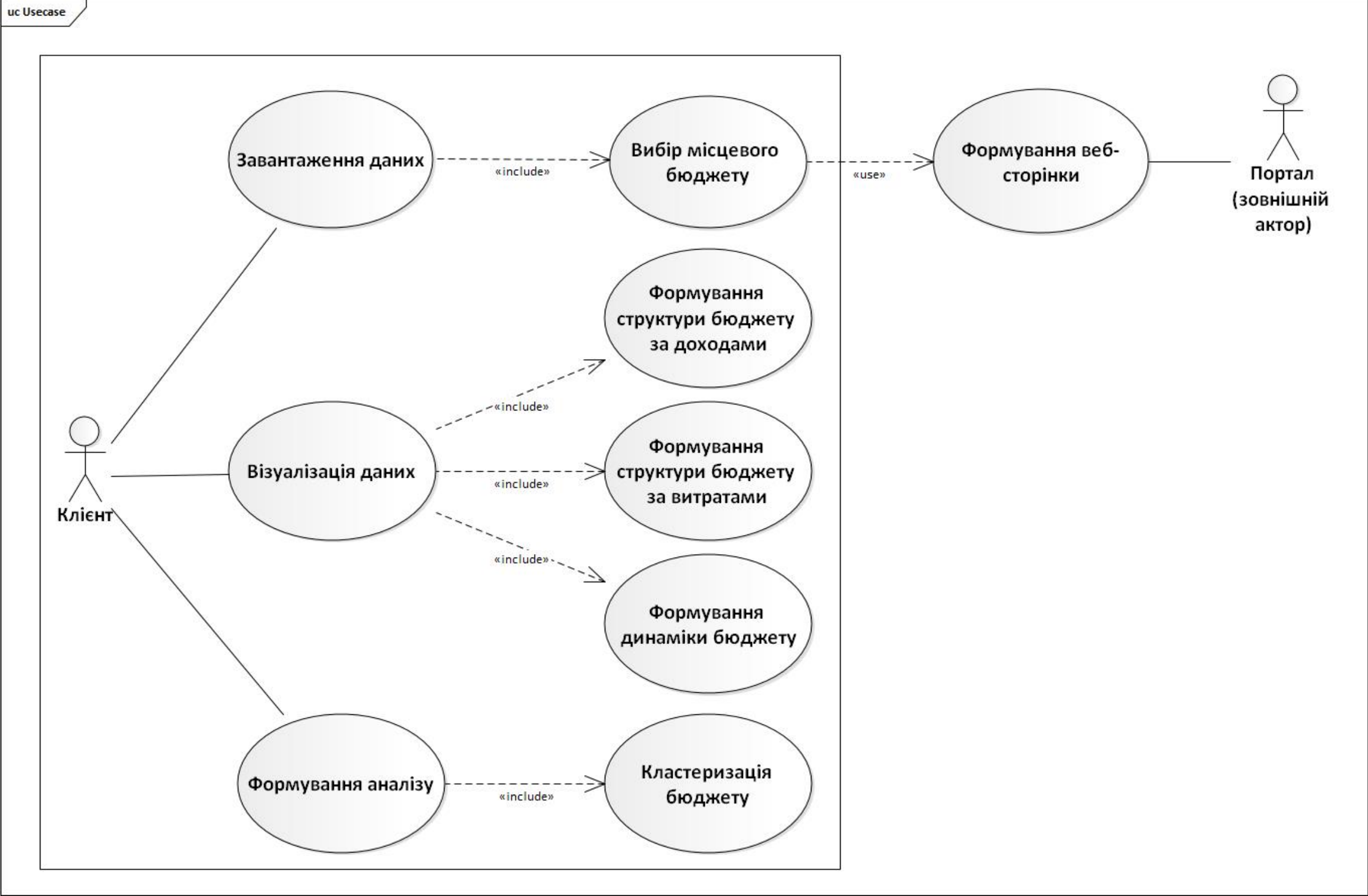
21. MongoDB Можливості. [Электронный ресурс]. – Режим доступа: <https://uk.wikipedia.org/wiki/MongoDB#Можливості>

22. Spending. [Электронный ресурс]. – Режим доступа: <https://spending.gov.ua/>

23. Spending: про портал. [Электронный ресурс]. – Режим доступа: <https://spending.gov.ua/about1>



					ДП ІС-5113.1181-с.СБД								
					Схема бази даних				Літера		Маса	Масштаб	
Зм.	Арк.	№ документа	Підпис	Дата									
Розробив		Коткова А.А.											
Перевірив		Олійник Ю.О.											
Т. кон.									Аркуш 1		Аркушів 1		
Н. кон.		Тєлишева Т.О.			Комплекс задач кластеризації даних місцевих фінансів				КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51				
Затвердив		Олійник Ю.О.											



					ДП ІС-5113.1181-с.ССВ			
					Схема структурна варіантів використання	Літера	Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата				
Розробив		Коткова А.А.						
Перевірив		Олійник Ю.О.						
Т. кон.					Комплекс задач кластеризації даних місцевих фінансів	Аркуш 1		Аркушів 1
Н. кон.		Телишева Т.О.				КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51		
Затвердив		Олійник Ю.О.						

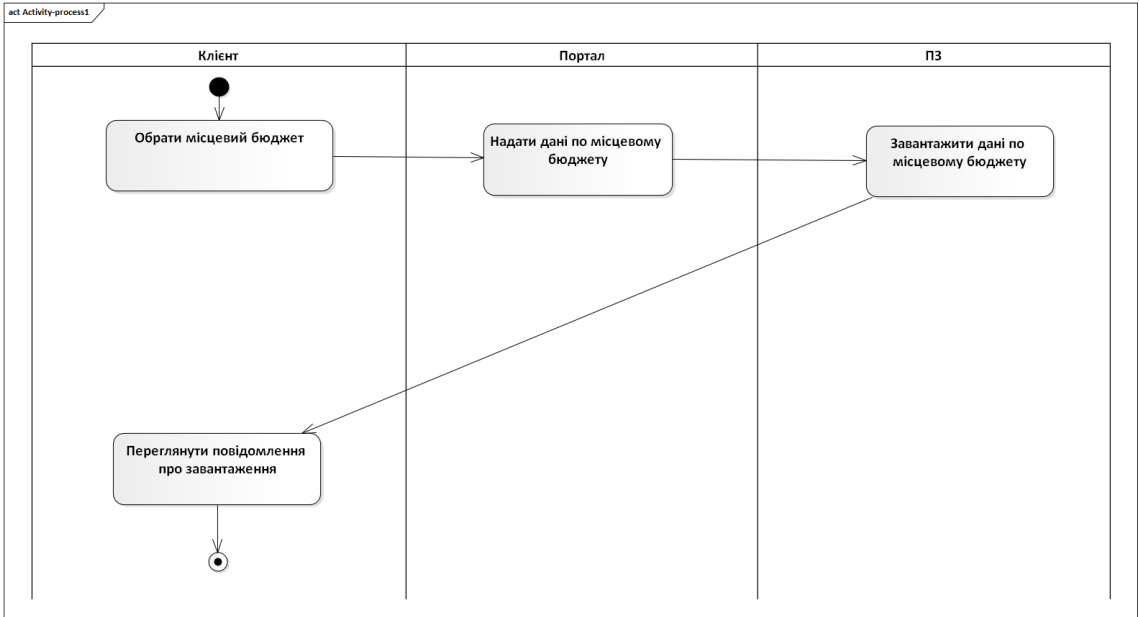


Схема структурна діяльності процесу завантаження даних по обраному місцевому бюджету

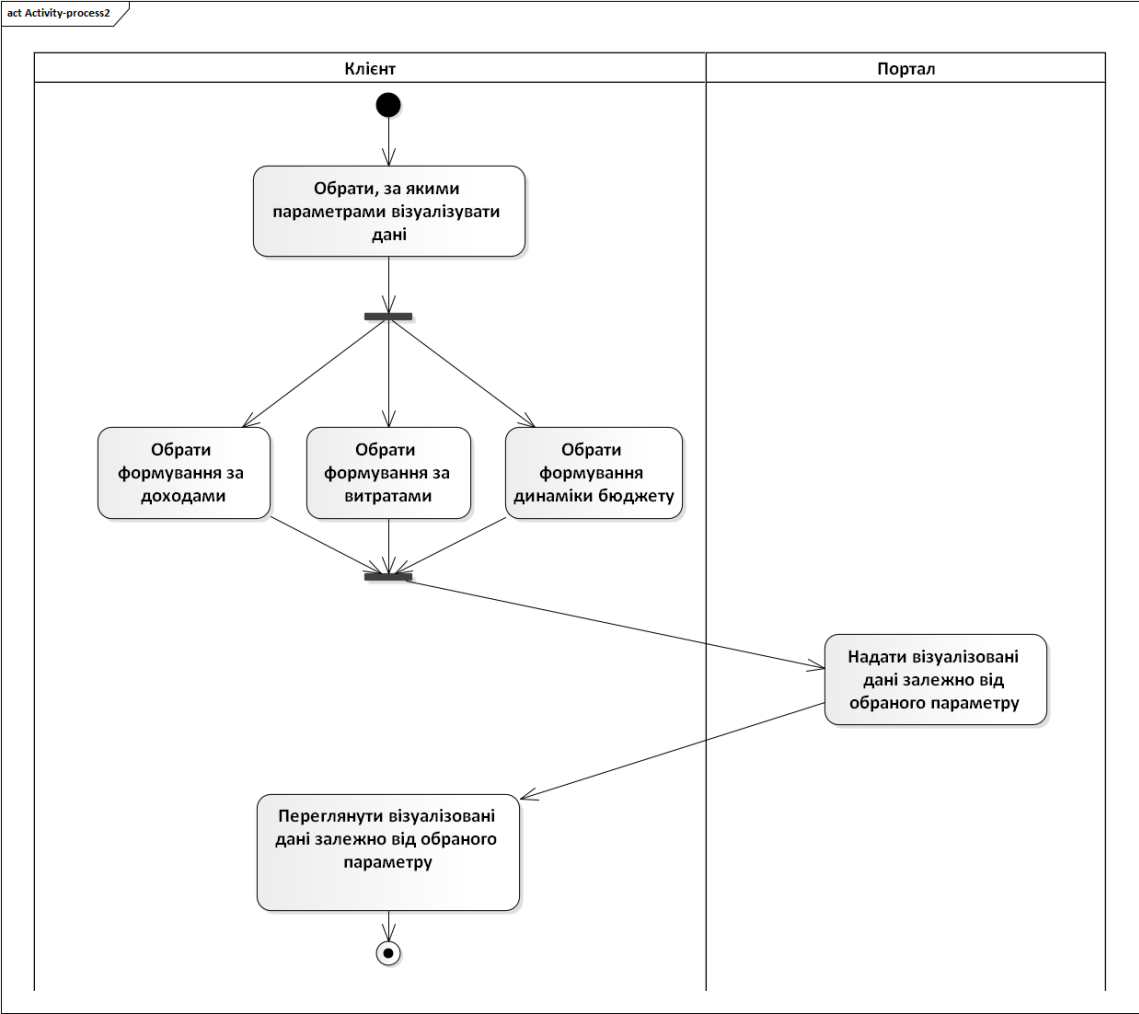


Схема структурна діяльності процесу візуалізації даних залежно від обраного параметру візуалізації

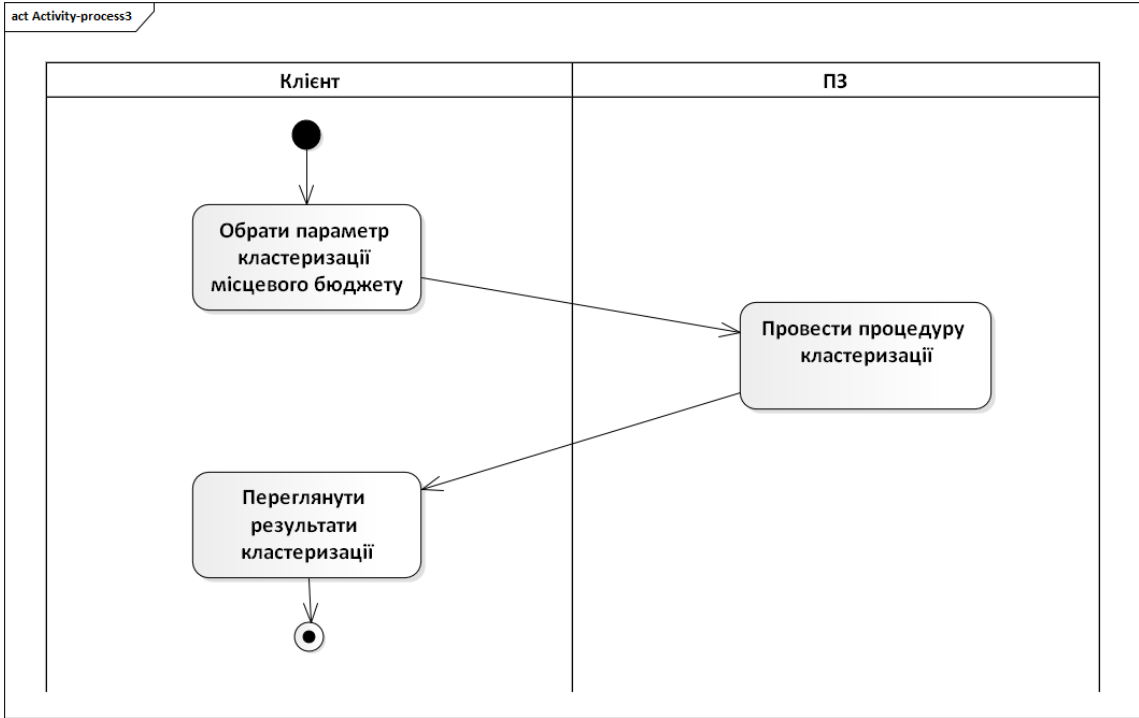
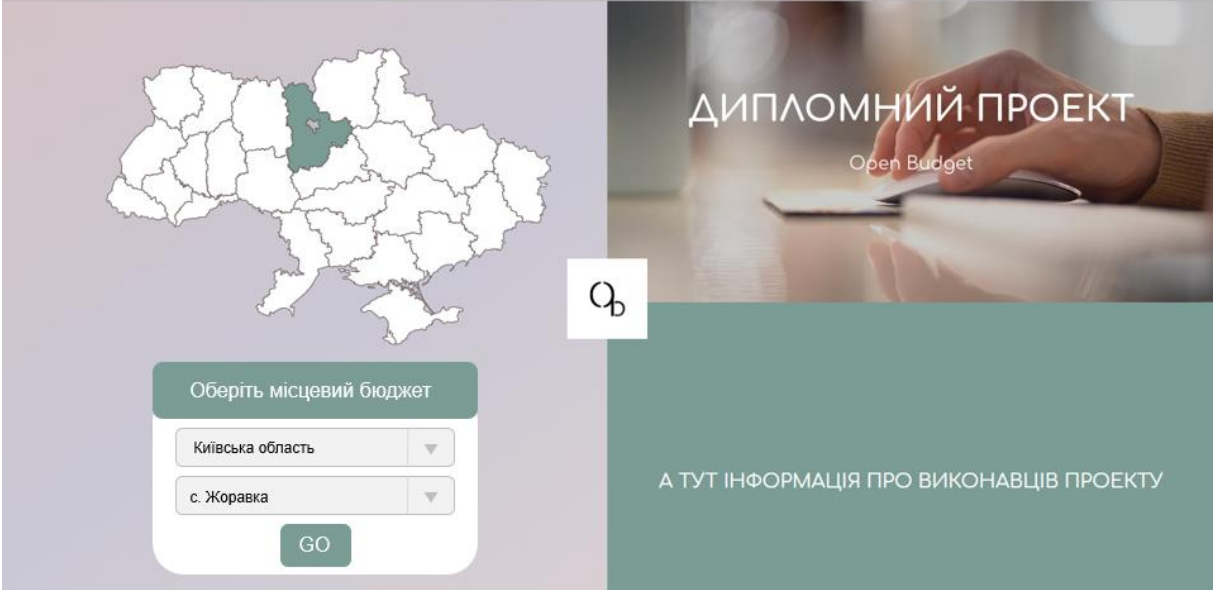


Схема структурна діяльності процесу кластеризації місцевих бюджетів

					ДП ІС-5113.1181-с.ССД						
					Схема структурна діяльності						
Зм.	Арк.	№ докум.	Підп.	Дата	Комплекс задач кластеризації даних місцевих фінансів						
Розроб.		Коткова А.А.									
Перев.		Олійник Ю.О.									
Т. Кон.											
					КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51						
Н. Кон.		Тєлишева Т.О.									
Затв.		Олійник Ю.О.									
					Лит.		Маса		Масштаб		
					Аркуш 1		Аркушів 1				



Екранна форма головної сторінки



Бюджет села ЖОРАВКА

Виконання бюджету

Структура бюджету

Рік: 2018 Місяць початку: Січень Місяць завершення: Березень Код: 10000000 **ПОШУК**

Код	Найменування	Розпис на рік	Виконано за період	Виконання до уточненого річного розпису
10000000	Податкові надходження	330 513.00	325 134.92	98.37
14000000	Внутрішні податки на товари та послуги	2 794.00	2 797.20	100.11

Екранна форма сторінки з інформацією про доходи



Бюджет села ЖОРАВКА

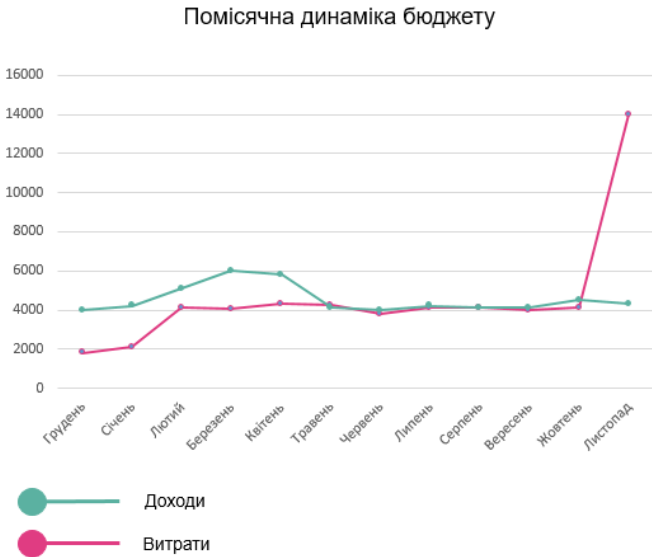
Виконання бюджету

Структура бюджету

Рік: 2018 Місяць початку: Січень Місяць завершення: Березень Код: Усі **ПОШУК**

КПК	КФК	Найменування КПК	Розпис на рік	Коштористні призначення на рік	Виконано за період	Виконання до уточненого річного розпису
0150	0111	Організаційно-аналітичне та матеріально-технічне забезпечення діяльності обласної ради, районної ради, районної у місті ради (у разі її створення), міської, селищної, сільської рад	305 255.00	305 255.00	80 459.75	26.35
6030	0620	Організація благоустрою населених пунктів	45 886.00	45 886.00	13 708.48	29.87

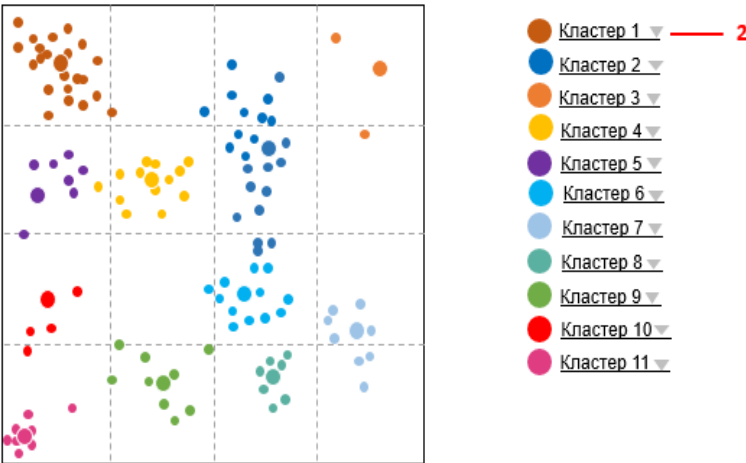
Екранна форма сторінки з інформацією про видатки



Екранна форма сторінки візуалізації динаміки бюджету



Оберіть кількість кластерів: 11 1



Екранна форма сторінки кластеризації бюджетів

					ДП ІС-5113.1181-с.КЕ							
					Креслення вигляду екранних форм	Літера			Маса		Масштаб	
Зм.	Арк.	№ документа	Підпис	Дата								
Розробив		Коткова А.А.										
Перевірів		Олійник Ю.О.										
Т. кон.												
					Комплекс задач кластеризації даних місцевих фінансів	Аркуш 1			Аркушів 1			
Н. кон.		Тєлишева Т.О				КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51						
Затвердив		Олійник Ю.О.										

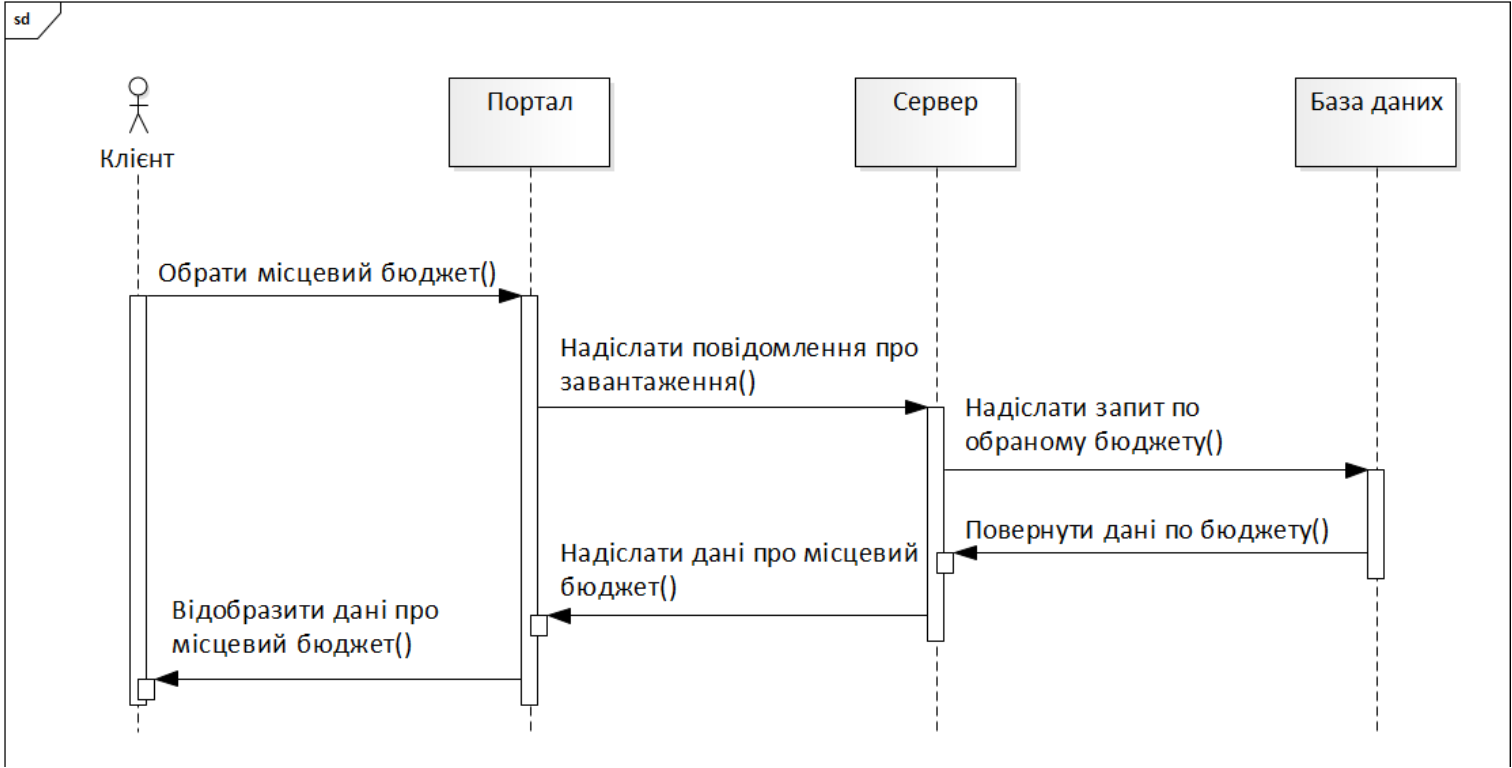


Схема структурна послідовності процесу завантаження даних по обраному місцевому бюджету

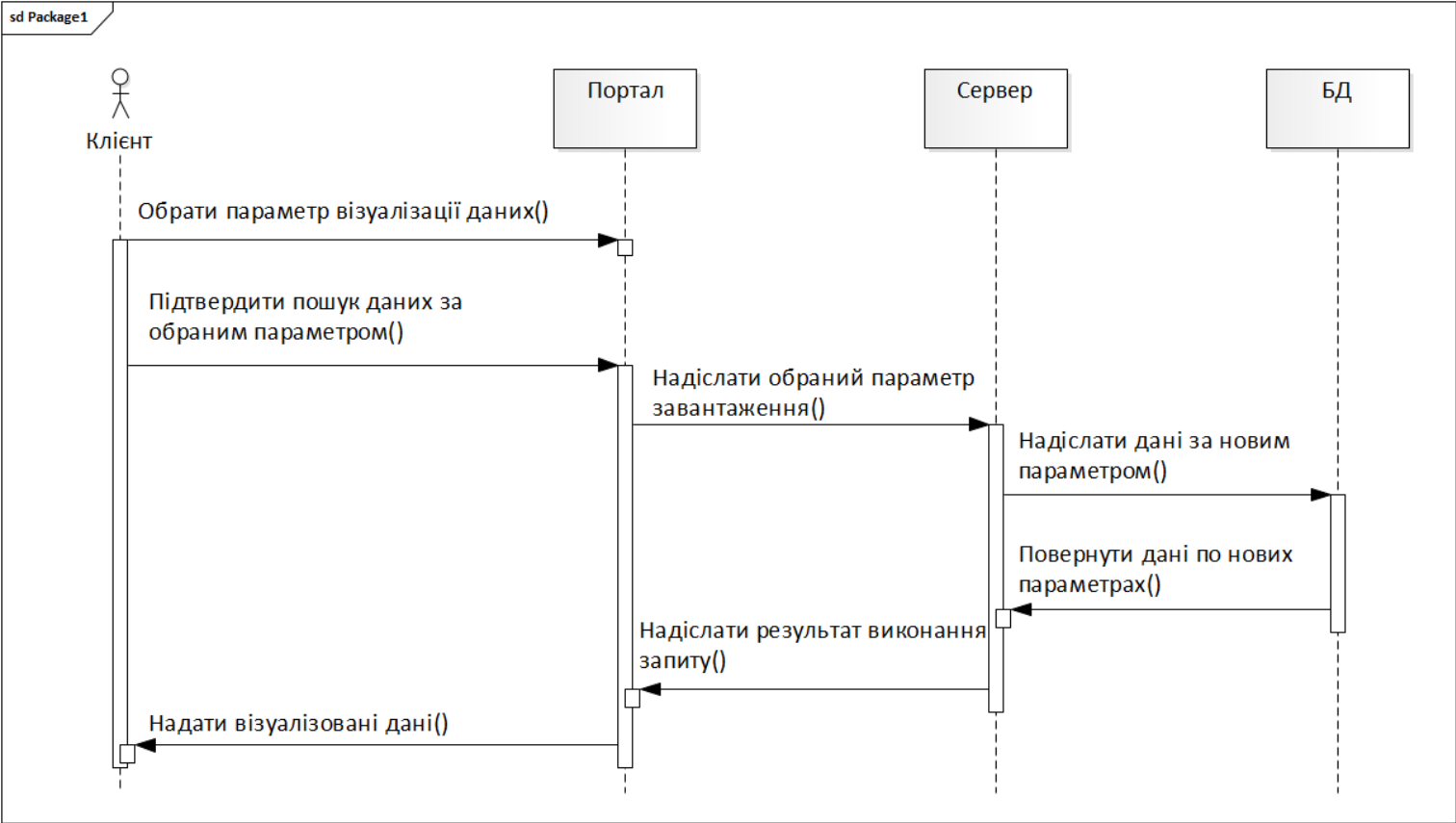


Схема структурна послідовності процесу візуалізації даних залежно від обраного параметру візуалізації

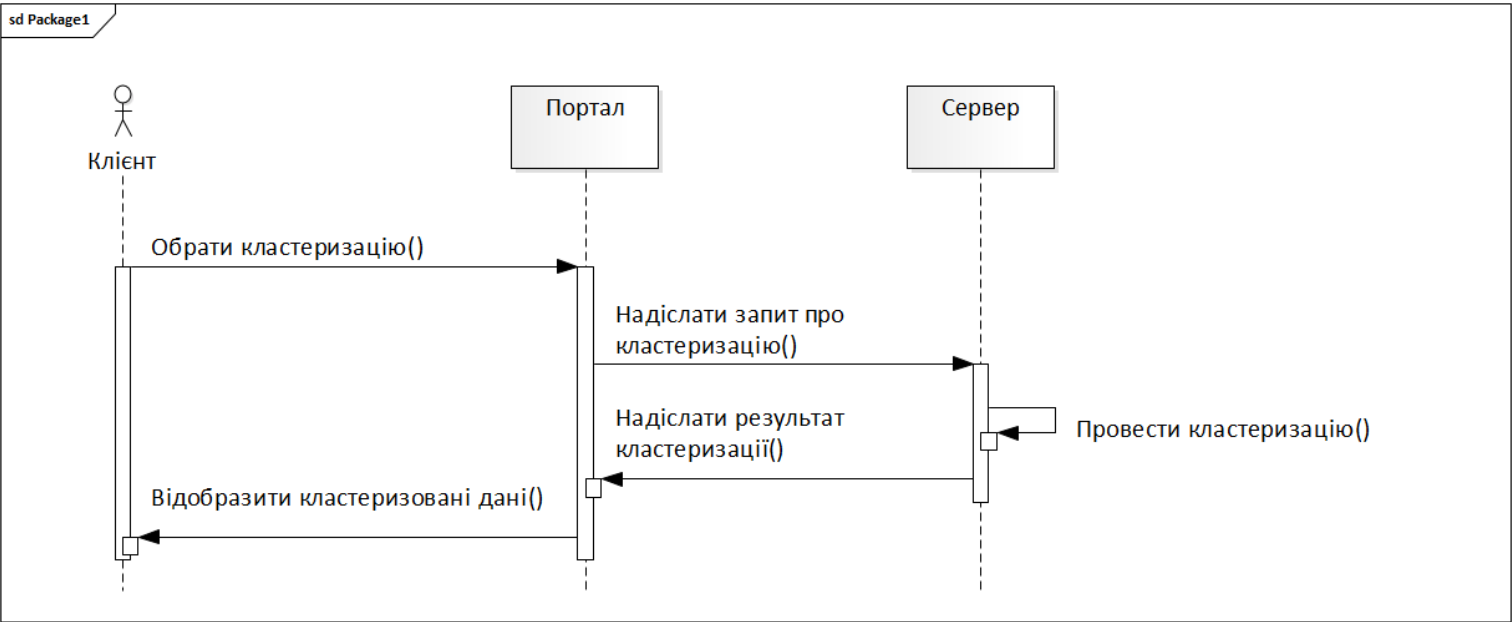
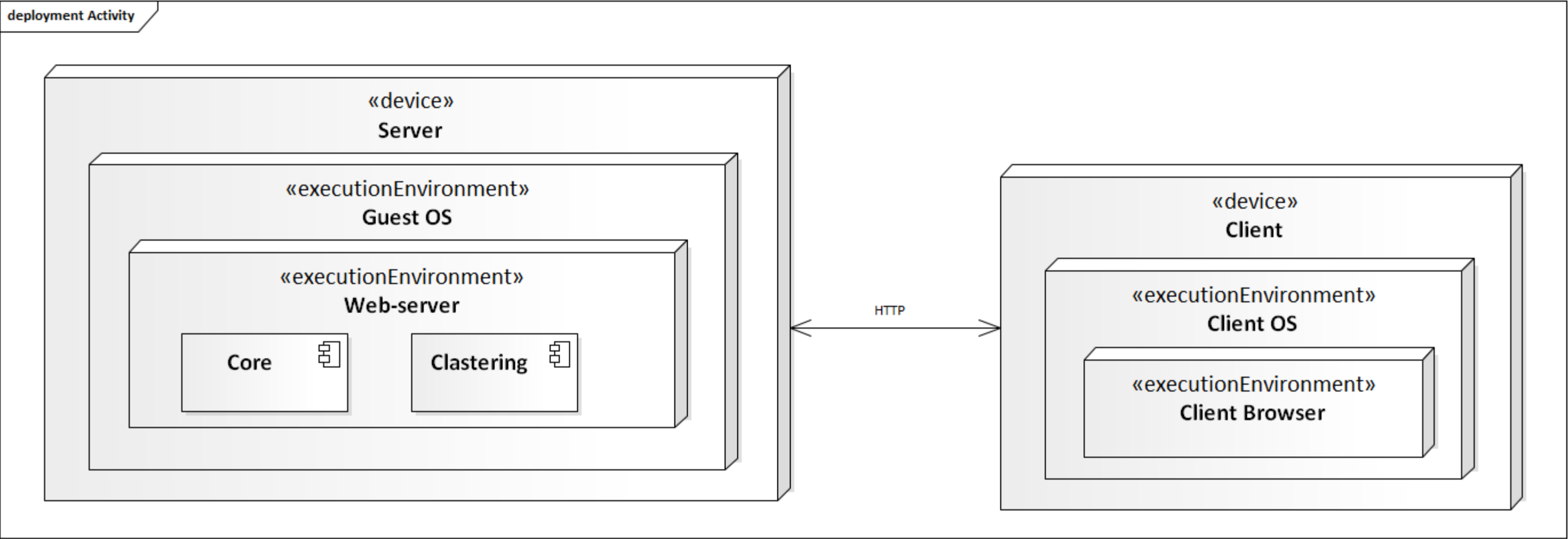


Схема структурна послідовності процесу кластеризації місцевих бюджетів

						ДП ІС-5113.1181-с.ССП					
						Схема структурна послідовності	Лит.		Маса	Масштаб	
Зм.	Арк.	№ докум.	Підп.	Дата							
Розроб.		Коткова А.А.									
Перев.		Олійник Ю.О.									
Т. Кон.							Аркуш 1		Аркушів 1		
						Комплекс задач кластеризації даних місцевих фінансів	КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51				
Н. Кон.		Тєлишева Т.О.									
Затв.		Олійник Ю.О.									



					ДП ІС-5113.1181-с.ССР			
					Схема структурна розгортання	Літера	Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата				
Розробив		Коткова А.А.						
Перевірив		Олійник Ю.О.						
Т. кон.					Комплекс задач кластеризації даних місцевих фінансів	Аркуш 1		Аркушів 1
Н. кон.		Телишева Т.О.				КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51		
Затвердив		Олійник Ю.О.						